NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

APPLYING ASYNCHRONOUS TRANSFER MODE TO THE MARINE CORPS BASE LEVEL INFORMATION INFRASTRUCTURE

by

Gregory T. Gdanski

June 1999

Thesis Co-Advisors: William J. Haga
John Osmundson

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE June 1999	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE : Applying Asynchronous Transfer Mode to the Marine Corps Base Level Information Infrastructure

5. FUNDING NUMBERS

6. AUTHOR(S)
Gdanski, Gregory T.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE
A

13. ABSTRACT *(maximum 200 words)*

This thesis reports the findings of a simulation comparing network architecture configurations in terms of interactions and performance in the face of varying traffic demand. It models the U. S. Marine Corps Base Level Information Infrastructure using a top-down approach. Extend® queuing theory modeling software was used to decompose the network model with a bottom-up approach to testing and integration. Feasible network configurations were identified and modeled under varying load parameters. Asynchronous Transfer Mode was found to be suited as a distribution protocol at the infrastructure levels of backbone and area distribution node. Fast Ethernet and Ethernet were found to be optimal at lower levels of infrastructure. Network design recommendations are made for network engineers.

14. SUBJECT TERMS
Systems, Networks, ATM

15. NUMBER OF PAGES
197

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

Approved for public release; distribution is unlimited

APPLYING ASYNCHRONOUS TRANSFER MODE TO THE MARINE CORPS BASE LEVEL INFORMATION INFRASTRUCTURE

Gregory T. Gdanski
Major, United States Marine Corps
B.S., University of Oklahoma, 1986

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL
June 1999

Author: Gregory T Gdanski

Gregory T. Gdanski

Approved by: William J Haga

William J. Haga, Thesis Co-Advisor

John S. Osmundson

John Osmundson, Thesis Co-Advisor

for

Reuben T. Harris, Chairman
Department of Systems Management

ABSTRACT

This thesis reports the findings of a simulation comparing network architecture configurations in terms of interactions and performance in the face of varying traffic demand. It models the U. S. Marine Corps Base Level Information Infrastructure using a top-down approach. Extend® queuing theory modeling software was used to decompose the network model with a bottom-up approach to testing and integration. Feasible network configurations were identified and modeled under varying load parameters. Asynchronous Transfer Mode was found to be suited as a distribution protocol at the infrastructure levels of backbone and area distribution node. Fast Ethernet and Ethernet were found to be optimal at lower levels of infrastructure. Network design recommendations are made for network engineers.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

A. PURPOSE

The purpose of this thesis is to conduct a comparative analysis of a variety of Base-Level Information Infrastructure (BLII) architectures in an effort to provide quantifiable guidance into the relative performances of each design.

B. BACKGROUND

As a network engineer I have observed substantial sustained growth in the local and wide area networks (LAN/WAN) within the Marine Corps since 1986. There has been a variety of protocols and information transmission media from which military network designers could choose when building or upgrading an installation's BLII. In the late 1980s, LANs were built within a building with outside connectivity generally limited to a few connections to other buildings within range of the existing copper telephony infrastructure. As technology improved, applications migrated into the LAN/WAN architecture. Network load evolved from simple electronic mail (Email) and on-line mainframe batch transactions to a variety of bandwidth-intensive applications such as Internet browsing, video teleconferencing (VTC), and distributed computing.

Information Technology (IT) managers responsible for maintaining the BLII have faced several challenges during the period of technological growth. These challenges relate to identifying specific design parameters needed to meet the IT needs of the command for the expected life cycle of the installed network infrastructure. Quantifying these design parameters is difficult. This is due to the rapid technological evolution of IT, the difficulty in gathering empirical data on existing IT utilization, and the unpredictable growth in tenant commands aboard the installation. A challenge in designing a BLII is selecting the correct transmission protocol(s) and bandwidth to meet the needs of an installation for the next ten to fifteen years.

Different standards bodies are responsible for copper/telephony standards, fiber optic standards, and radio frequency (RF) standards. These organizations include the Institute of Electrical and Electronic Engineers (IEEE), the International Telecommunications Union Telecommunications Standardization Sector (ITU-TSS), and

many others. Each standard body works in a focused technological area with limited interaction with the other standards organizations. A further complication is the need to integrate copper, fiber and RF equipment within a BLII and the availability and cost of network load analysis tools and applications. Each manager has had to rely on personal knowledge, experience and crude estimation techniques to determine which architecture to install. This may result in either over-engineering or under-engineering causing a premature need for network redesign and rework. Either of these situations may result in a waste of scarce financial resources.

IT managers at the installation level have lacked formal guidance from higher levels in the respective service's Command, Control, Communications, Computers and Information (C4I) structure. This lack of guidance is partially attributable to the same challenges faced by the installation's IT manager. Compounding the service C4I structure's ability to address the issue is the great disparity between installations. Within the Marine Corps, each installation varies in terms of geographical layout, numbers of personnel and computers, and even the applications that are used within the installation.

There is no one answer that is best for every situation. What is needed is an analysis based on different levels of network traffic load to show how the different configurations perform relative to each other. IT managers need some source to compare the performance characteristics of different architectures in order to make an informed decision as to the appropriate BLII for their situation.

In this thesis I have performed a comparative analysis of a variety of BLII architectures by modeling different configurations using a queuing theory application software called Extend. The model allows variation of BLII transmission protocols and bandwidths, as well as varying levels of network traffic, and performances are measured in terms of data latency.

C. OUTLINE OF REMAINING CHAPTERS

The modeling hardware and software environment is covered in Chapter II. Decomposition of the problem and issues regarding the design of the system are in Chapter III. Chapter IV details the development of the model, software challenges and solutions, and identifies the configurations to be modeled. Chapter V provides detailed analysis of the model runs and in-depth comparisons of the configurations with varying network traffic loads. The final conclusions and recommendations are in Chapter VI.

II. MODEL DEVELOPMENT ENVIRONMENT

A. INTRODUCTION TO EXTEND MODELING SOFTWARE

The model was developed using Extend version 4.0, a queuing theory application software produced by Imagine That!, Inc. The application centers around standardized libraries of process objects, referred to as blocks. It is designed to be a user friendly way to facilitate the rapid development of queuing systems by dragging and dropping blocks from the library to the model workspace. Models can be built to simulate continuous flow problems or discrete events. The different libraries that are built into the application are generally designed specifically for one or the other, however, many objects within the libraries are interchangeable with either type of model.

The two main types of blocks are item blocks and attribute blocks. Item blocks receive and process discrete events or items that pass through them. Attribute blocks receive and process attribute values associated with items, although the items do not specifically transit through these blocks. Blocks have item or attribute connectors associated with them. The flow of the model is determined by the order of the connections between blocks of the model.

Figure 1 illustrates the connectivity within an Extend model. An item follows the flow from left to right, through the Get Attribute block, the Queue block, and into the Select Path block. An attribute from the item is compared to a constant attribute, resulting in a decision on which path to take out of the Select Path block.

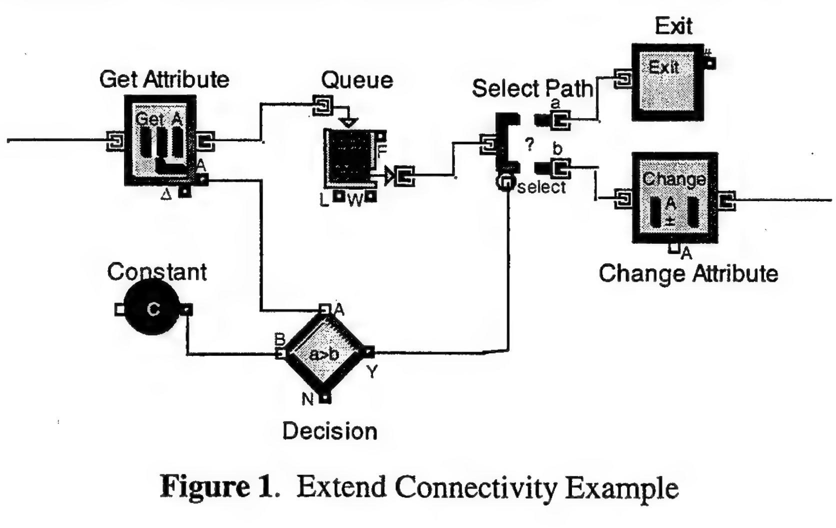


Figure 1. Extend Connectivity Example

Blocks can be grouped together as a hierarchical block and represented as a custom block in the model workspace. This facilitates the logical collection of similar or related functions into a single representation in the workspace. Internal connections needed for the internal blocks are built into the custom block. By this manner, complex models can be condensed at the highest level into a readily understandable design. The specific configuration of a hierarchical block can be seen by drilling down. Hierarchical blocks can be built by combining hierarchical blocks into further hierarchical abstractions.

B. COMPUTING ENVIRONMENT

1. Initial Computer Configuration

Development of the model began with a Dell Dimension model XPS M200s Pentium computer. The PC was configured with a 200 MHz processor, 64 Mb RAM, 512 Mb of virtual memory running Microsoft Windows 95. Development of the low level packet handling hierarchical blocks was easily accomplished. However, the process of building higher level hierarchical blocks exceeded the ability of the PC. This occurred when the size of the model exceeded 109 Mb, reflecting approximately one eighth of the overall model size.

2. Secondary Computer Configuration

Development of the model switched to a Dell Dimension model XPS R400 Pentium II computer. The PC was configured with a 400 MHz processor, 256 Mb RAM, 1.5 Gb of virtual memory, and running Microsoft Windows NT. This PC was used for the remainder of the model development, however, when the size of the model exceeded 210 Mb, the application itself became degraded to the point that adding or connecting each additional block required several seconds of wait time. The model was redesigned so that the size was reduced from approximately 900 Mb to 68 Mb. Implementation of the model redesign is addressed in a Chapter IV.

C. MODEL DESIGN METHODOLOGY

In order to minimize the actual development time of the model, a detailed paper model was built to identify key modular objects that would be essential to replicating a generic Base-Level Information Infrastructure (BLII), determining essential hierarchical blocking of logical groups of objects, and identifying unique functional objects necessary but not available in the pre-defined Extend libraries. By completing a detailed design on paper prior to actual coding, initialization procedures and item attributes could be defined and included at the start of the coding, thereby minimizing the amount of redesign necessary as the components were built and integrated.

1. Paper Design Model

The initial paper model design was completed using a top-down approach. This facilitated a logical, systematic decomposition of the complex nature of the base networking system. The design methodology focused on developing complete objects that applied equally to every level of the BLII model and which were compatible with the variety of protocol configurations that would be modeled. The intent was to build all of the generic objects as high-level, hierarchical blocks and install them in a custom Extend library. Any changes that were required as the model was built and tested could be accomplished by updating a single custom library block and replicating it throughout the model. Customization that was necessary due to location within the BLII would be integrated external to these blocks. This primarily would involve setting traffic attributes such as BLII source and destination addresses. The paper design of the model is Appendix A.

2. Base-Level Information Infrastructure (BLII) Structure

Each LAN, WAN, and base topology is essentially unique. The amount of processing delay associated with transmitting traffic through a network is referred to as its latency. In order to model and test the network flow for data latency, a design was chosen that reflected a standardized configuration of LAN and WAN architectures within the BLII. While the configuration does not replicate any specific base, post, or station, the intent is to analyze the architectures and protocols in general and not to identify an ideal

architecture for all locations. This model reflects only performance characteristics *within* the base and does not address inter-base performance or connectivity issues.

The generic BLII architecture chosen is based on a single point of entry into the BLII for all incoming traffic. Similarly, all traffic exiting the BLII follows the same path. Shared network resources are located centrally on the base network backbone. This creates, in effect, a network server farm for all services shared communally. These shared services would include items such as the base Web server, mail services, application file servers, shared CD-ROM tower libraries, and other similar types of services. The design chosen still allows for network file servers to exist within the local workgroups and traffic patterns established at that level take the increased traffic load into account.

In addition to the network services, the design included a direct point-to-point connection for a high-end video-teleconferencing (VTC) suite within the organization. Because of the bandwidth required for full-motion video, voice, and data, the design only included a single suite. The suite can be used for three of the major applications requiring this level of quality; namely, command VTC, classroom distance learning, and virtual staffing.

Distribution to the rest of the base is accomplished through area distribution nodes (ADN) tied into the backbone ring. These represent geographical clusters of buildings that are tied into the backbone through a high-speed fiber optic switch. For purposes of consistency, each ADN provides connectivity for seven buildings. Distribution within each building is accomplished through internal distribution via three distinct switch locations or wiring closets. Each of these locations represents either a separate floor or wing of a building and is detailed in the model as a workgroup. The workgroups model the network traffic associated with the equivalent of roughly 72 ports of end-user computing equipment. This architecture is represented in Figure 2.

3. Network Traffic Flow

The focus of the model was to evaluate the flow of information through the BLII architecture and compare the relative performance of different configurations. To get the true interactions between various traffic items, the initial design modeled packet level movement of data. This included converting items into appropriate protocol transport formats such as IP datagrams, Ethernet frames, and ATM cells. This was important because packets moving throughout the BLII mingle from the various sources and better

replicate the differences in transit time when small traffic items are competing for network resources with large traffic items.

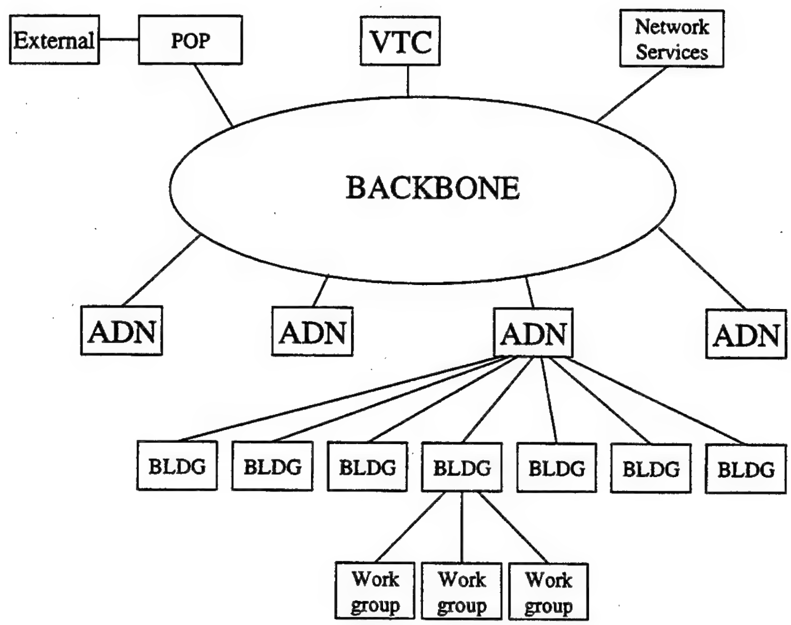


Figure 2. BLII Architecture

Eighty percent of the initial design of the model consisted of the packet handling routines. Overall model growth to the limits of Extend's capabilities required a high-level redesign using item movement through the BLII without decomposition. In implementing this new design, estimation on the packet level interactions had to be used to account for the detailed waits and delays that would have been captured at the more granular level.

III. METHODOLOGY

A. RESEARCH FOCUS AND APPROACH

1. Research Focus

The focus of designing and running the model was to look at how the various protocols and BLII architectures impacted the throughput of information. In order to identify specific configurations that could be recommended, based solely on performance characteristics, the model had to accurately represent the various types and volumes of traffic that would normally be present within a Marine Corps organization. In addition, the relative performances would need to be contrasted with varying levels of network traffic workload.

2. Research Approach

The approach chosen in implementation of this model was to focus on the latency of those network applications that represented real-time applications of information transfer. The two key traffic types that met that requirement were both voice/video applications. The first was desktop-to-desktop video teleconferencing (VTC). The second was full-motion voice video applications that would be seen at a command level VTC full-motion video suite. Performance statistics were gathered and analyzed from these two applications in preparing the conclusions and recommendations.

Capturing empirical data latency was accomplished by accumulating two primary time components as a traffic item passed through the BLII. The first component is wait. Wait is the amount of time an item is queued in a buffer while no processing is actually taking place with respect to that specific item. The second component is delay. Delay is the amount of time that an item is held for direct processing and is broken into two elements. *Process delay* is the processing time associated with handling the item within the electronic switching and routing equipment. It included conversions between protocols and packet collection that would occur at the point at which those conversions would be required within the BLII. *Transport delay* is the time associated with actually transporting the item between BLII levels and is tied to protocol and bandwidth

specifications of the architecture configurations. Combined, these elements capture the overall latency associated with traffic movement.

B. MEASURES OF EFFECTIVENESS

Measuring the relative performance of the different architectures was with three measures. Each configuration was run three times. The measures for each run were gathered and averaged for the analysis values.

1. Overall Latency Average For Entire Model Run

The first measure used in the analysis is the overall average for all voice and video traffic that completed movement through the model. It represents the sustained rate of performance for the traffic through the entire model run. This value only represents the amount of average latency encountered within the BLII architecture and does not make any estimation of latency at the distant installation or during the transit between installations.

2. Number of Critical Latency Spikes

The second measure used in the analysis of the various configurations was the number of times that the average latency exceeded 0.05 seconds during the $1/10^{th}$ of a second sampling interval. The $1/20^{th}$ of a second critical latency value was selected prior to the beginning of the model runs because it represented a reasonable point at which the latency becomes noticeable to the end user.

The selection of 0.05 was arbitrary. However, in viewing the results of each model run it became evident that selecting a different criterion value would not have greatly changed the overall model results. The same ratio of spikes between compared configurations remained if the critical latency value was reduced to either 0.045 or 0.04 seconds. The same proportionality also remained if the critical latency value was increased to 0.06 but, when raised to 0.07, began to skew the overall performance sharply towards configurations using ATM.

Each spike represents an average delay that was longer than 0.05 seconds for all measured traffic during a single $1/10^{th}$ of a second interval. The number of spikes only identifies those delays attributable to the BLII architecture and does not include any

estimation of latency associated with the distant installation's infrastructure or the transit between installations.

3. Maximum Critical Latency Spike

The third measure used was the maximum 1/10th of a second average for each model run. This value reflects the relative maximum latency that was observed, thereby providing a worst-case consideration. The critical latency spike only identifies the greatest latency attributable to the BLII architecture and does not include any latency associated with the distant installation's infrastructure or the transit between installations.

C. MODEL PARAMETERS

The model was designed to simplify the ease with which the user can change the configurations being tested. For comparative analysis between model runs, only one variable or set of variables is changed. Each change is implemented in a single location.

1. BLII Architecture

In order to model different architectures, the parameters that identify the protocol and bandwidth of the transport at each level are set within a single block at the top level of the model. The numeric code corresponding to each specified transport protocol and associated bandwidth is identified in Table 1.

Value	Transport Protocol	Transport Bandwidth
0	FDDI	100 Mbps
1	ATM OC-12	622 Mbps
2	ATM OC-3	155 Mbps
3	ATM DS-3	45 Mbps
4	IP	100 Mbps
5	Gigabit Ethernet	800 Mbps
6	Fast Ethernet	100 Mbps
7	Ethernet	10 Mbps

Table 1. Architecture Parameters

The five attribute values that are set in the Start block are the protocols for the External, Backbone, ADN, Building and Workgroup levels of the BLII. As part of the initialization process executed at the beginning of each model run, those attributes are passed through each layer of the BLII model, setting the table lookup values at the appropriate locations.

2. Time Simulation

The time simulation used for the model run was 10 seconds. This reflects a snapshot of performance at peak workload times. This parameter is initialized on the main Extend pull-down menus prior to beginning the simulation. Run times being simulated can be easily changed. However, the length of time required to run the model changes proportionally with the change in time simulated. Using 10 seconds as the simulated time, the normal medium, and heavy network load simulations ran at a consistent time of 45, 62, and 82 minutes respectively.

3. Traffic Types

The model included fourteen types of traffic generators to simulate the variety of applications expected within a Marine Corps network. Each generator was built using random number generators to establish the frequency and size of each item of traffic. For each type of traffic, separate distribution profiles were established for frequency and size. The distributions and parameters used in the model generators are in Appendix F. The following are the types of generators used at various levels in the architecture:

- Email without attachment
- Email with attachment
- Command full-motion VTC suite
- Desktop VTC application
- Inter-device communication
- Network management application
- Internet NIPRNET/SIPRNET traffic
- Internet File Transfer Protocol (FTP) setup
- Internet FTP downloads
- Internet commercial World Wide Web (WWW) traffic going out

- Internet commercial WWW traffic coming in
- Network resource request
- Response to network resource request
- Network security

4. Network Load Analysis

Network load analysis was accomplished by developing a profile for each type of traffic that would exist in the network and its respective size and frequency parameters. In addition to the specific traffic generators, a sensitivity analysis generator was added to the workgroup level traffic block.

The parameters for this additional generator were set so that no additional traffic would be generated during the 10 second normal network load runs. The parameters of this block were increased to simulate a fixed rate of increased overall traffic level for each respective load increase model. By this manner, refining the specific changes of individual generators was not necessary and an easily quantifiable increase was readily assimilated. To change the load parameters, the user must open one of the Sensitivity Analysis blocks using the Open Hierarchical option from the main task bar, make the changes to the parameters as outlined by the directions within the block, and then save the block using the Update All option. This applies the changes to every Sensitivity Analysis block within the model.

a. Normal Network Load

The normal network load consisted of all of the standard traffic type generators. The normal network load is the baseline from which the additional load increases were tested. Each workgroup within the BLII generated traffic from identical generators with identical size and frequency distribution parameters. The Sensitivity Analysis generator had its parameters set so that no additional traffic would be introduced into the run. Actual traffic generated from within each workgroup would still vary between each workgroup due to the randomness built into each generator.

Empirical data for each type of generator was not available. In lieu of empirical data, best-guess approximations were used based on the author's personal experiences as the Information Systems Management Officer for a Marine Corps Base. While these estimations will not be accurate for every installation and may not be

accurate for any specific installation, they do represent an aggregate level of network load that could be expected within an installation that has migrated many of its normal functions onto the IT infrastructure.

b. Medium Network Load Increase

To test the performance of the various architectures under workload increases, the parameters of the normal workload traffic generators were left unchanged. A Sensitivity Analysis generator was used to simulate a steady level of increased traffic that remained within the BLII system. This increased traffic "noise" is intended to represent a relative increase of any combination of traffic that could occur on the network. The added traffic travels up from the workgroups in one ADN, transits through the backbone and proceeds down to the workgroup level of an adjacent ADN. The traffic level generated at each workgroup level by the Sensitivity Analysis generator is 400 kbps. The additional effective traffic congestion generated is in Table 2.

c. Heavy Network Load Increase

Comparing the relative performances of different configurations using a heavy workload increase were implemented in the same manner as the medium workload increase. The traffic level generated at each workgroup level by the Sensitivity Analysis generator is 600 kbps. The additional effective traffic congestion generated is in Table 2.

Level	Medium Load	Heavy Load
Backbone	33.6 Mbps	50.4 Mbps
ADN	16.8 Mbps	25.2 Mbps
Building	2.4 Mbps	3.6 Mbps
Workgroup	0.8 Mbps	1.2 Mbps

Table 2. Effective Traffic Congestion Levels

D. ANALYSIS STRATEGY

The analysis of the three measures of effectiveness for each configuration would be compared to extract relative performance patterns for the key transmission types that were being evaluated. The main focus is on comparing the distribution between the

backbone, area distribution nodes and buildings. It is at this level that the aggregation of network traffic has the greatest impact. For all but one architecture configuration the external connectivity was fixed at classic IP. The one external configuration that was not classic IP was set as an ATM connection. For all but two of the architecture configurations the workgroup distribution was fixed at Ethernet. The two workgroup configurations that were not Ethernet were set as an ATM connection.

IV. MODEL DEVELOPMENT

A. BASE NETWORK ARCHITECTURE

The first step in developing a base network model is determining which general topology to follow. The second step is determining the scale of the organization. No two Marine bases, posts, or stations are identical in either size or layout. Taken together, the general topology and the organizational scale determine the architecture layers within the organizational network. Within and interconnecting the layers are the networking protocols that can be employed. The architecture chosen for this model assumes a large base dispersed into 28 buildings in each of four geographical clusters. This does not represent any specific Marine base, but rather approximates a possible base configuration that is large enough that the importance of real-time data latency can be accurately reflected.

1. Architecture Layers

The generic base architecture that was chosen for the model included the following five layers.

a. External to the BLII

This represents the sources of traffic entering into the BLII, the traffic exiting the BLII, and the basic point of presence that each of these items must pass through. This point of presence can be a router, firewall, POP server, or similar device or devices. This layer feeds directly into the base backbone. For purposes of this model, connections are 100 Mbps at a minimum.

b. Backbone

The backbone of the network architecture is the heart of any BLII network. It includes all centralized, shared, network resources such as file servers, service servers, and unique or low-density components such as CD-ROM towers. In addition, for this model, it includes a command-level VTC suite. This suite, while not necessarily co-located with the shared network resources, would have a direct, point-to-point connection

from the backbone. This suite would be used as a combination studio for applications such as Command VTC, Distance Learning Classroom, or Virtual Staffing Office.

Networking components included as part of the backbone include the primary distribution system using fiber optic cable and an appropriate array of high-speed routers and/or switches appropriate for each configuration being modeled. At a minimum, the backbone would have a speed of 100 Mbps. Distribution of information from the backbone through the geographic areas is accomplished through four area distribution nodes. The types of traffic that are generated at this level include responses to network service requests; network management applications; command VTC; device-to-device traffic between components such as routers, servers, and switches; and network security traffic.

c. Area Distribution Node (ADN)

The ADNs represent data distribution within a small geographical region of the base. It consists of routers and/or switches controlling the distribution within the area. Each ADN provides network connectivity to seven buildings. For purposes of modularity, each ADN was built identical. While no base has identically configured nodes, this method was chosen to provide the best workload analysis of the various architectures. The only type of traffic generated at this level is the device-to-device traffic between components such as routers and switches.

d. Building Distribution

Each building includes switches and routers used to connect to the backbone via the ADNs. For modularity and equivalent workload analysis, each building was developed with three internal distribution groups known as workgroups. These can represent data closets on separate floors of a building or in separate wings of single-story buildings, both of which are common configurations within Marine buildings. Again, while there are substantial variations in the architectures within buildings, identical configurations were chosen to best represent workload traffic. The only type of traffic generated at this level is the device-to-device traffic between components such as routers and switches.

e. Workgroup Distribution

Each workgroup represents the clustering of end-user computing equipment centralized from a single wiring closet. It includes traffic generators that replicate the items from the PCs and devices within the workgroup. Traffic types include the following:

- Electronic mail (Email) without attachments
- Email with attachments
- Requests for network services
- Internet File Transfer Protocol (FTP) setup
- Commercial World-Wide Web (WWW) "surfing"
- NIPRNET/SIPRNET messages
- Network device-to-device traffic
- Desktop VTC applications
- Network Management applications

The workgroup is the source for the majority of the traffic that is generated within the BLII. Although it is normal for at least one workgroup to also exist directly off of the backbone to support the IT personnel, one was not included in this model. The model is capable of being modified to include a workgroup. However, it is equally likely that the main backbone location also serves as an ADN and that one of the buildings connected into the ADN is the IT services building where the main backbone switching components are co-located.

2. Network Protocols

The model includes the following protocols as options that can be selected for the various levels within the BLII architecture. Only those possible configurations that logically made sense were tested, although the model could run any configuration chosen.

a. Classic Internet Protocol (IP)

Classic IP is the baseline for the model. IP represents the Network Layer in the Open Systems Interconnection (OSI) model. This was used to represent the connection external to the BLII. Its speed was set at 100 Mbps to ensure that the exiting path would not represent such a bottleneck that the remainder of the analysis was undistinguishable.

While that speed is substantial, readers must also be aware that, at least within the Marine Corps, most installations currently are limited to a range from fractional T-1 speeds of 256 or 512 kbps to multiple T-1 lines totaling 3.0 or 4.5 Mbps. This disparity between the model and reality is indicative of another major challenge, that of gaining access to adequate bandwidth outside of the BLII to meet current needs. However, that challenge is outside the scope of this thesis. Nonetheless, the 100 Mbps minimum is indicative of likely future demands for external bandwidth.

All remaining protocols used by the model represent the Transport Layer in the OSI model. As such, they encapsulate the IP packets or datagrams created at the network layer. For this reason, IP is modeled with a relative efficiency of 1.0 as the baseline protocol.

b. Fiber Distributed Data Interface (FDDI)

FDDI is the oldest fiber optic transport protocol. FDDI is implemented with dual, counter-rotating rings with an operating speed of 100 Mbps. Although there is little Copper Distributed Data Interface (CDDI) infrastructure within Marine Corps installations, the same parameter will represent either, the major difference in capability being the distance limitations associated with copper and the susceptibility of copper to RF and power interference.

c. Asynchronous Transfer Mode (ATM)

ATM transport speeds used by the model include 622 Mbps (OC-12), 155 Mbps (OC-3), and 45 Mbps (DS-3). These three speeds represent the Synchronous Optical Network (SONET) speeds that apply to ATM. The Marine Corps has not formally adopted SONET/ATM as an infrastructure standard. However, through informal communications within the IT community of the Marine Corps, there is a *de facto*

consensus that ATM networks should be SONET capable in anticipation of future migration. For this reason, the higher ATM speeds of 1.2 Gbps (OC-24) to 9 Gbps (OC-192) and beyond were not incorporated as model options.

d. Ethernet

Ethernet transport speeds used by the model include 10 Mbps (Ethernet), 100 Mbps (Fast Ethernet), and 800 Mbps (Gigabit Ethernet). Ethernet is the prevalent transport protocol throughout the Marine Corps. The most recent configuration to evolve is Gigabit Ethernet which has identical characteristics to the other configurations with the exception of a reduced inter-frame gap.

B. DEVELOPING AND TESTING

1. Bottom-Up Development

Unlike the top-down designing phase, the model was built through bottom-up development. The lowest level components were built first. Generic utility blocks that would be needed at various non-related points were built and tested, then included in the functional hierarchical blocks as needed. Because the paper model decomposed the entire BLII system down to packet-level processes, development of these packet-level objects was completed first. As objects were individually built and tested, they were integrated into higher level functional objects. As these more abstract objects were tested and validated, they too were then incorporated into blocks with a broader functional scope.

Through this process, the testing and debugging of the model components became relatively easy. A block that failed to perform as expected was analyzed only at the highest level, since the functionality of the lower level blocks had already been proven through detailed testing. This bottom-up approach facilitated the quick development of the overall model with the highest level of confidence in the underlying processes and procedures.

2. Libraries

One of the key functionalities of Extend is its implementation of block libraries. A model developer can create custom libraries to hold developed hierarchical blocks for

use in other places within the model or within separate models. This ability to quickly and easily reuse code facilitated the object-oriented approach to designing this model. As lower-level hierarchical blocks were validated, they were installed in custom libraries. Whenever a parameter within a library block required changes, only one block had to be modified. Upon saving the changes to that block, an option is available to replicate the changes to the library and all identical blocks in the model. This simplified the implementation of changes.

C. EXTEND LIMITATIONS

During the development of the model, several challenges arose that directly related to the way the Extend application software was designed. As issues emerged, the author contacted the Technical Support Section of Imagine That! at its San Jose headquarters. The application developers worked closely with the author to remedy problems that arose. In spite of the timely assistance provided by Imagine That!, some of the challenges for which workarounds were provided were not entirely resolved. While none of the problems caused Extend to outright fail, several of the problems were serious enough that when combined they required a high level redesign of the model.

1. Model Size Growth

Designing and integrating all of the packet handling processes created the first indication of a size problem. All packet functionality was located in one hierarchical block, namely, the Data Pipe. While it tested fully functional, the Data Pipe is a core function at every level of the BLII architecture. Each workgroup is built around a Data Pipe. By moving up one level, each building contains a total of four. Similarly, each ADN contains a total of 29. Each Data Pipe comprised approximately 6 Mb of Extend code. The aggregation of just this one key hierarchical block came to over 700 Mb of Extend code.

Integrating the components up to the building level caused memory problems on the initial workstation used in developing the model. At this point, the application and the model were moved to the second workstation, which had greater memory capacity and processing speed. Continuing the integration up to the ADN worked until it included the core ADN functions and five of the seven building hierarchical blocks. At that point, the model represented approximately one-fifth of the overall model size and contained

210 Mb of Extend code. At this point the application ran so slowly that it became impractical to continue with the development of the model with the low-level design.

2. Application Efficiencies

A second limitation that directly related to the size of the model is the internal processing efficiency. One feature of Extend is the recalculation of the model sequencing after each new block is connected into the model. This recalculation is completed from start to finish after each connection is made. This recalculation is not noticeable on small models. When the model size grew to larger than 80 Mb, the processing delay became noticeable, although it was not recognized as such at the time.

The problem occurred when any input occurred from the keyboard or mouse prior to completion of the recalculation. Any input from an I/O device caused the application to crash, closed down Extend entirely, and loosing any changes made since the last save. The Technical Support Section of Imagine That! was able to identify the source of the problem but will not implement the solution until the next version of Extend which incorporates some of the needed design changes.

3. Virtual Memory

An initial problem that occurred during development directly related to the amount of RAM and virtual memory allocated. Programs that run sequentially use pagination to swap unused sections of the program out of RAM to make space for needed sections of the program during execution. Whenever the program size is larger than the available RAM needed to load it, pagination will occur. Because of the BLII design, virtually every section of the program contains an active portion of the model at every given time step of the simulation. This caused extensive pagination throughout each developmental test run, as evidenced by run times in excess of four hours for each 10-second simulation. Additionally, saving changes to a hierarchical block such as the Data Pipe routinely took over two hours to implement when incorporated globally in the model.

D. IMPLEMENTATION CHANGES

With the limitations identified in Extend, the overall scope and design of the model had to be adjusted to fit within those limits. The critical implication from making

these changes was that the model, although still providing an accurate comparison of different architectures, lost some of the accuracy of interaction between packets of different traffic types and sizes.

1. Item Versus Packet Modeling

Replacement of the lower level packet handling blocks with more generic estimation blocks was essential to reduce the overall size of the model. Rather than timing the arrivals between first and last packets, collecting and assembling packets for conversion to another transport protocol, and having an effective method for simulating full buffers and lost packets, generic estimation equations were used to accumulate the approximated values.

2. Change Implications

By implementing the changes in the overall functionality of the model, some of the critical interactions between packets were lost. Of note is that when dealing with the packet level processes, packets from different traffic items arrive interspersed with traffic from other sources. This directly affects the queuing order and transit times associated with each item. Because the packets intermingle, smaller traffic items work their way through the system rapidly, while larger traffic items take longer to process. By converting to item level processing, only one item can be in transit through a data pipe at any given time. This has the effect of slowing down smaller items and speeding up larger items.

Because of this loss of interaction, the overall network performance less accurately replicates the complex real-world network system. However, the same level of degradation would apply equally to every configuration modeled. For this reason, the results of the model are still valuable as an analysis tool.

E. MODEL RUN MATRIX

In determining which model configurations to run, architectures were selected that reflected possible migration paths of legacy designs. The premise was used that as the architecture moved from a lower level to a higher level, bandwidth would be greater than or equal to the lower level. Not doing this would cause a data choke point to occur.

Fourteen configurations were chosen to represent likely architectures for the modeling of normal network load. These configurations are in Table 3.

Configuration	External	Backbone	ADN	Building	Workgroup
01	IP	Fast E/N	Fast E/N	Fast E/N	E/N
02	IP	Gb E/N	Fast E/N	Fast E/N	E/N
03	IP	Gb E/N	Gb E/N	Fast E/N	E/N
04	IP	FDDI	Fast E/N	Fast E/N	E/N
05	IP	FDDI	Gb E/N	Fast E/N	E/N
06	IP	FDDI	FDDI	Fast E/N	E/N
07	IP	ATM OC-12	ATM OC-3	Fast E/N	E/N
08	IP	ATM OC-12	ATM OC-12	Fast E/N	E/N
09	IP	ATM OC-3	ATM OC-3	Fast E/N	E/N
10	IP	ATM OC-12	ATM OC-3	ATM OC-3	E/N
11	IP	ATM OC-12	ATM OC-3	ATM OC-3	ATM DS-3
12	IP	ATM OC-12	ATM OC-12	ATM OC-3	ATM DS-3
13	ATM OC-3	ATM OC-12	ATM OC-3	Fast E/N	E/N
14	IP	IP	IP	IP	E/N

Table 3. Model Configurations

The configuration designators remain constant throughout all model runs. However, only those configurations that performed well at the basic network load level were further modeled at the medium workload increase level. Likewise, only selected configurations from the medium workload increase runs were further tested at the heavy workload increase level. The decision as to the number and types of configurations that would proceed to the next level of modeling was made after all configurations for the previous level were completed and analyzed.

The final model is presented in the appendices. Because of the size, the model was broken into four sections. These sections are the BLII architecture blocks, general utility blocks, functional blocks, and traffic generator blocks. These sections are in Appendices C through F respectively.

V. MODEL RESULTS

A. NORMAL NETWORK LOAD

For the normal network load analysis, all fourteen initial configurations were modeled. Each configuration was run three times with the three measures from each run averaged. The detailed results of all runs are in Appendix B. The averaged results for each configuration is in Table 4. For the correlation of configuration identifier with the protocols used at each level, refer to Table 3. Comparing the various configurations focused on groups of similar architectures. This focus was on the three key layers of the BLII where congestion will be the greatest. These layers are the backbone, the connections between the ADN and the buildings, and the connections between the building and the workgroups.

Configuration	Average Latency	Number of Spikes > 0.05 Seconds	Maximum Latency
01	0.0172	8.33	0.1856
02	0.0169	10.33	0.0988
03	0.0157	7.33	0.1305
04	2.6958	79.67	5.6581
05	2.2837	98.00	4.6770
06	0.2802	64.00	0.8694
07	0.0144	5.33	0.1018
08	0.0143	6.00	0.0997
09	0.0138	5.67	0.0976
10	0.0136	6.33	0.1353
11	0.0132	6.33	0.0732
12	0.0125	7.00	0.0827
13	0.0131	5.67	0.0797
14	0.0153	9.67	0.1135

Table 4. Normal Network Load Results

1. Gigabit Ethernet/Fast Ethernet Configurations

Configurations 01 through 03 represent architectures that are pure Ethernet protocol. All three configurations performed with similar average latency, the range being 0.0157 to 0.0172 seconds. The frequency with which the latency exceeded the 0.05 seconds value ranged from 7.33 to 10.33 times. There was a substantial variation between the configurations when looking at the maximum latency delay that was encountered, with values in the range of 0.0988 to 0.1856 seconds. The highest value was higher than expected and could possibly be a chance occurrence attributed to the probabilistic nature of the random number generators for the one specific run. In general, the performance improved slightly as Gigabit Ethernet replaced Fast Ethernet. Overall, each of these three configurations could be expected to perform adequately given normal network load parameters.

2. ATM/Fast Ethernet Configurations

Configurations 07 through 09 represent architectures that have ATM as the backbone and ADN levels with Fast Ethernet used between the building and workgroups. All three configurations performed at nearly identical levels. The average latency only varied from 0.0138 to 0.0144 seconds. The frequency with which the latency exceeded the 0.05 seconds value ranged from 5.33 to 6.00 times. The maximum latency delay only varied from 0.0976 to 0.1018 seconds. In general, no quantifiable difference was noticed between the three ATM/Fast Ethernet architectures. All could be expected to perform adequately given normal network load parameters.

3. Homogeneous ATM Configurations

Configurations 10 through 12 represent architectures that are purely ATM protocol for the three key layers. All three configurations performed at nearly identical levels. The average latency only varied from 0.0125 to 0.0136 seconds. The frequency with which the latency exceeded the 0.05 seconds value ranged from 6.33 to 7.00 times. The maximum latency delay varied from 0.0732 to 0.1353 seconds. This variation was greater than expected and could possibly be attributed to the probabilistic nature of the random number generators for the one specific run. Two of the three runs had maximum latency values lower than those of the ATM/Fast Ethernet configurations. In general, no

quantifiable difference was noticed between the three ATM architectures and all could be expected to perform adequately given normal network load parameters.

4. Normal Network Load Summary

All of the Ethernet and ATM configurations summarized above could be expected to adequately support the network load represented by this model. There were quantifiable levels of improvement between the three categories as ATM was introduced lower into the architecture. The most noticeable improvement was between the homogeneous Ethernet architectures and the mixed ATM/Fast Ethernet architectures. There was only nominal improvement between the ATM/Fast Ethernet architectures and the homogeneous ATM architectures. The slight improvement seen would probably not be sufficiently cost effective to justify a homogeneous ATM network.

B. MEDIUM WORKLOAD INCREASE

For the medium workload increase analysis, seven configurations were modeled. Each configuration was run three times with the three measures from each run averaged. The detailed results of all runs are in Appendix B. The averaged results for each configuration are in Table 5. For the correlation of configuration identifier with the protocols used at each level, refer to Table 3.

Configuration	Average Latency	Number of Spikes > 0.05 Seconds	Maximum Latency
01	0.0234	8.33	0.1049
02	0.0232	9.33	0.1044
03	0.0200	9.00	0.1051
07	0.0151	4.33	0.0676
08	0.0165	5.33	0.0750
09	0.0158	5.33	0.0697
12	0.0149	1.33	0.0602

Table 5. Medium Workload Increase Results

1. Gigabit Ethernet/Fast Ethernet Configurations

Configurations 01 through 03 represent architectures that are pure Ethernet protocol. All three configurations performed with similar average latency, the range being 0.0200 to 0.0234 seconds. This represents an increased average latency range of 0.0043 to 0.0063 seconds. The most notable performance was the configuration with the most Gigabit Ethernet. The frequency with which the latency exceeded the 0.05 seconds value ranged from 8.33 to 9.00 times which represented a nominal increase in the number of latency spikes. There was virtually no variation between the configurations when looking at the maximum latency delay that was encountered, with values in the range of 0.1044 to 0.1051 seconds. In general, the performance improved as Gigabit replaced Fast Ethernet. Overall, each of these three configurations could be expected to perform adequately given medium workload increase parameters.

2. ATM/Fast Ethernet Configurations

Configurations 07 through 09 represent architectures that have ATM as the backbone and ADN levels with Fast Ethernet used between the building and workgroups. All three configurations performed at nearly identical levels. The average latency only varied from 0.0151 to 0.0165 seconds. This represents an increased average latency range of 0.0007 to 0.0022 seconds. The frequency with which the latency exceeded the 0.05 seconds value ranged from 4.33 to 5.33 times, indicating a nominal improvement. The maximum latency delay only varied from 0.0676 to 0.0750 seconds. This trend shows a substantial improvement in performance over the Normal Network Load model runs. In general, no clearly quantifiable difference was noticed between the three ATM/Fast Ethernet architectures. All could be expected to perform adequately given medium workload increase parameters. Even so, configuration 07 performed the best of all three configurations when looking across all three measures of effectiveness.

3. Homogeneous ATM Configurations

Configuration 12 represents an architecture that is purely ATM protocol for the three key layers. This configuration performed comparable to the Normal Network Load run with an average latency of 0.0149 seconds, an increase of 0.0024 seconds. This increase brought it within range of the ATM/Fast Ethernet configurations. Of note is the

reduction in the number of times the latency exceeded the 0.05 seconds. This average value dropped from 7.00 to 1.33 times, indicating a substantial improvement. This configuration also reflected a reduction in the maximum latency delay from 0.0827 to 0.0602 seconds. This anomaly may be reflective of the probabilistic nature of the random number generators for one specific run. In general, this ATM architecture could be expected to perform adequately given medium workload increase parameters.

4. Medium Workload Increase Summary

All of the Ethernet and ATM configurations summarized above could be expected to adequately support the network load represented by this model. Notably, the Ethernet configurations degraded at a greater rate than did those containing ATM in terms of average latency. There was substantially better improvement by the ATM architectures over the homogeneous Ethernet architectures when looking at both the number of times the latency delay exceeded the 0.05 seconds value as well as the maximum latency observed. There was little quantifiable difference between the ATM/Fast Ethernet architectures and the homogeneous ATM architecture. Again, the slight improvement seen would probably not be sufficiently cost effective to justify a homogeneous ATM network over an ATM/Fast Ethernet network.

C. HEAVY WORKLOAD INCREASE

For the heavy workload increase analysis, five configurations were further modeled. Each configuration was run three times with the three measures from each run averaged. The detailed results of all runs are in Appendix B. The averaged results for each configuration is in Table 6. For the correlation of configuration identifier with the protocols used at each level, refer back to Table 3.

Configuration	Average Latency	Number of Spikes > 0.05 Seconds	Maximum Latency
02	0.0425	31.33	0.1239
03	0.0369	14.67	0.1342
07	0.0362	23.00	0.0809
09	0.0399	28.33	0.1117
12	0.0384	19.00	0.0934

Table 6. Heavy Workload Increase Results

1. Gigabit Ethernet/Fast Ethernet Configuration

Configurations 02 and 03 represent architectures that are pure Ethernet protocol. All three configurations performed with substantially different average latency, the range being 0.0369 to 0.0425 seconds. This represents an increased average latency range of 0.0169 to 0.0193 seconds. The most notable performance again was the configuration with the most Gigabit Ethernet. The frequency with which the latency exceeded the 0.05 seconds value ranged from 14.67 to 31.33 times which represented a substantial increase in the number of latency spikes. There was much less variation between the configurations when looking at the maximum latency delay that was encountered, with values in the range of 0.1239 to 0.1342 seconds. In general, the performance improved as more Gigabit Ethernet replaced Fast Ethernet. Overall, both of these configurations reflected performance levels indicative of sustained degradation. It is not clear if these configurations could provide adequate support for a network under these parameters. By far the best performing Ethernet architecture included Gigabit Ethernet on the backbone as well as for distribution from the ADN to the building and using Fast Ethernet distribution within the building to the workgroups.

2. ATM/Fast Ethernet Configuration

Configurations 07 and 09 represent architectures that have ATM as the backbone and ADN levels with Fast Ethernet used between the building and workgroups. Both configurations reflected greater variance across all three measures of effectiveness than with the normal or medium workloads. The average latency varied from 0.0362 to

0.0399 seconds. This represents an increased average latency range of 0.0211 to 0.0241 seconds. The frequency with which the latency exceeded the 0.05 seconds value ranged from 23.00 to 28.33 times, indicating a substantial increase. The maximum latency delay varied from 0.0809 to 0.1117 seconds. This trend shows a substantial degradation for configuration 09 over the Normal Network Load model runs, with less degradation for configuration 07. It is not clear if configuration 09 could provide adequate support for a network under these parameters. However, configuration 07 still represents a degraded but probably acceptable level of performance.

3. Medium Workload Increase Summary

All of the Ethernet and ATM configurations summarized above could not be guaranteed to adequately support the network load represented by this model. Only configuration 07 shows a propensity to withstand heavy workload increases without excessive degradation. As the workload increased, the configurations with less ATM converged in terms of performance with the higher configurations using Gigabit Ethernet.

D. ACROSS THE SPECTRUM

Throughout the model runs, there was a clear indication that the performance of categories of architectures containing ATM outperformed those with homogeneous Ethernet. As the workload increased, the disparity in performance between the categories became more noticeable. The variations in performance between ATM/Fast Ethernet and homogeneous ATM configurations was nominal. This indicates that the use of ATM down to the workgroup layer of the BLII architecture provides no noticeable improvement, largely because the congestion does not exist at that level where high-speed networking pays off. Overall, the best performing architecture was configuration 07. This configuration included ATM OC-12 on the backbone, ATM OC-3 for distribution from the ADN to the buildings, Fast Ethernet distribution within the building to the workgroups, and Ethernet distribution within the workgroup.

VI. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The use of ATM technologies in the BLII produces noticeable and relevant advantages in performance for real-time applications involving voice and video over purely Ethernet architectures. This advantage is more noticeable the higher the relative usage rate of the available bandwidth. While there was improvement in performance for each additional level of the BLII that included ATM, the relative improvement between each incremental step was nominal and certainly not noticeable to an end-user. The best overall performance increases occurred when ATM was used as the distribution architecture for the backbone and the area distribution nodes.

As the workload on the network increased, every configuration modeled showed signs of degradation. It is important to note that the level of degradation was consistently less for architectures that included ATM as part of the central distribution.

At medium workload increases Ethernet configurations displayed average latency in the 0.0200 to 0.0234 seconds range while ATM configurations ranged from 0.0149 to 0.0165 seconds. In addition to this, the maximum latency delay encountered for Ethernet configurations was between 0.1044 to 0.1051 seconds while the ATM architectures ranged from 0.0602 to 0.0750 seconds.

At heavy workload increases Ethernet configurations displayed average latency in the 0.0369 to 0.0425 seconds range while ATM configurations ranged from 0.0362 to 0.0399 seconds. In addition to this, the maximum latency delay encountered for Ethernet configurations was between 0.1239 to 0.1342 seconds while the ATM architectures ranged from 0.0809 to 0.1117 seconds.

Only at the heavy workload increase did the lowest bandwidth configuration of ATM converge near the performance statistics of the highest Ethernet configuration. Also of importance is that the performance of homogeneous ATM networks compared to mixed ATM/Fast Ethernet networks was virtually identical for all workloads modeled. This indicates that the use of ATM down to the lowest levels of the BLII architecture will be difficult to justify when analyzed using Cost-Benefit Analysis.

Networks that are designed and installed with substantial excess capacity of bandwidth will work effectively with acceptable data latency using pure Ethernet

configurations. As long as relative volume to capacity ratio remains small, no noticeable degradation will be apparent to the end-user. The most robust architecture found across the entire spectrum of workloads was configuration 07. This configuration included ATM OC-12 on the backbone, ATM OC-3 for distribution from the ADN to the buildings, Fast Ethernet distribution within the building to the workgroups, and Ethernet distribution within the workgroup. This configuration is shown in Figure 3.

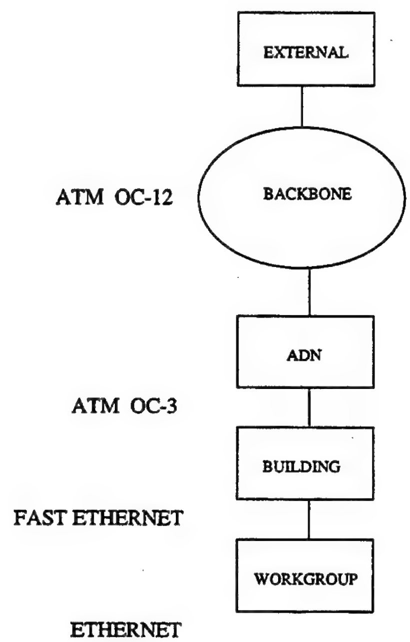


Figure 3. Recommended BLII Architecture

B. RECOMMENDATIONS FOR DOD

1. Use of Asynchronous Transfer Mode (ATM) Technology

The use of ATM technologies in the BLII is justified for adoption as the most effective and efficient distribution standard for garrison architectures in terms of overall network performance. Increases in the number and types of real-time and near real-time applications will further necessitate a flexible architecture that can handle different priorities of traffic. ATM provides a more consistent level of performance for these types of traffic in a network environment that can fluctuate substantially minute by minute or hour by hour. This model shows ATM to be the most appropriate distribution protocol

among the ones included in the model when looking solely at performance characteristics; however, the extent to which ATM technology is implemented within a specific base, post, or station cannot be determined solely by the model.

2. Use of Modeling Application Software

Continued use of modeling application software is highly recommended for research and analysis of existing and emerging technologies and processes. With the current inherent restrictions within Extend version 4.0, I would not recommend this software for detailed development and analysis of extremely complex systems of systems. Other applications such as CommNet and OpNet are more suited to these types of problems than Extend. However, for analysis of abstract processes and systems where performance parameters can be readily defined and reasonably estimated, Extend is very effective. The application itself is user friendly, reliable, and ideal for quick development of queuing theory applications.

C. SUGGESTED FURTHER STUDIES

1. Applying These Results to Determine DoD or USMC standards

The Extend model created for analyzing the performance of various network architectures effectively identified trends in technology and protocol performance based on baseline traffic and changes in network workload. These results, taken alone, should not be the basis for finalizing a standardized network architecture standard for DoD or the Marine Corps. Two key areas should be further addressed before a decision is made. A Cost-Benefit Analysis should be completed for the various architectures tested with this model. Additionally, new technology that is presently under development and being employed in limited locations also shows potential for application within the BLII.

a. Cost-Benefit Analysis (CBA)

This research specifically addressed the relative performance of alternative configurations of the BLII. The conclusions and recommendations provided are directly tied to those model results. Before any final determination on an optimal BLII

configuration standard can be made, a full CBA must be conducted on the comparative costs of the networking components associated with each configuration.

An additional CBA comparison that should be looked at directly relates to the volume to capacity ratio. It is possible that investing is a pure Ethernet architecture that is based on multiple connections between major network distribution points, each connection being either Fast Ethernet or Gigabit Ethernet, could have a lower per-port cost than an equivalent ATM architecture. While certainly less efficient, the possibility of a better overall CBA final result cannot be discounted offhand.

b. Internet Protocol over Dense Wavelength Division Multiplexing (IP/DWDM)

As this model was being developed, a new technology was emerging that primarily applied to continental inter-base distribution of information. IP/DWDM is a router-based distribution over multiple fiber optic cables, each having a capacity of OC-24 (1.244 Gbps) and greater. Although this technology is extremely new and still under development, the potential for a DoD-wide distribution technology is very good. With that capability, there is a strong possibility that this technology could also have applicability to the backbone distribution within the BLII as a natural extension of the global architecture.

2. Model Enhancement and Further Testing

Due to the constraints and inefficiencies of Extend, early design plans using analysis at the packet level had to be abandoned. During the development of the lower level hierarchical blocks, the packet processing routines were thoroughly tested and validated. The point at which the model exceeded the ability of Extend to effectively run it consisted of a fractional area distribution node containing only two of the eight planned buildings. At this stage, the model reached a size of 210 Mb and required over six hours of run time to simulate ten seconds of traffic. Removing the packet handling processes dramatically reduced the size and run time of the model, but at some cost to the accuracy of how the individual packets of different pieces of network traffic interacted. While every effort was made to accurately replicate the overall affect of these interactions in the more abstract, higher level design, it certainly remains a worthwhile endeavor to attempt

to continue with the packet level design of the BLII or some major sub-component. Should future releases of Extend prove to have enhanced capabilities or improved processing efficiencies, resumption of packet-level development is recommended.

The packet level portions of the model that were developed in the early stages were not incorporated into the final working model. These blocks were, however, retained. The Extend blocks for this portion of the model development have been included as Appendix G.

BIBLIOGRAPHY

"ATM Pocket Guide", revision B, TEKELEC, July, 1994.

DoD ATM Addressing Plan, version 1.0, Defense Information Systems Agency, 1998.

DoD ATM Standards, version 1.0, Defense Information Systems Agency, 1998.

DoD Defense Information System Network (DISN) Asynchronous Transfer Mode (ATM) System Specification, version 1.2c, Defense Information Systems Agency, 1998.

Duncan, Michele A., *An Analysis of Bandwidth Requirements for Collaborative Planning*, Master's Thesis, Naval Postgraduate School, June 1998.

Extend User's Manual, version 4, Imagine That, Inc., 1997.

Migrating to High Speed Networking, revision 1.0, American Research Group, 1996.

"Networking Protocol Reference Guide", revision 2.0, American Research Group, 1993.

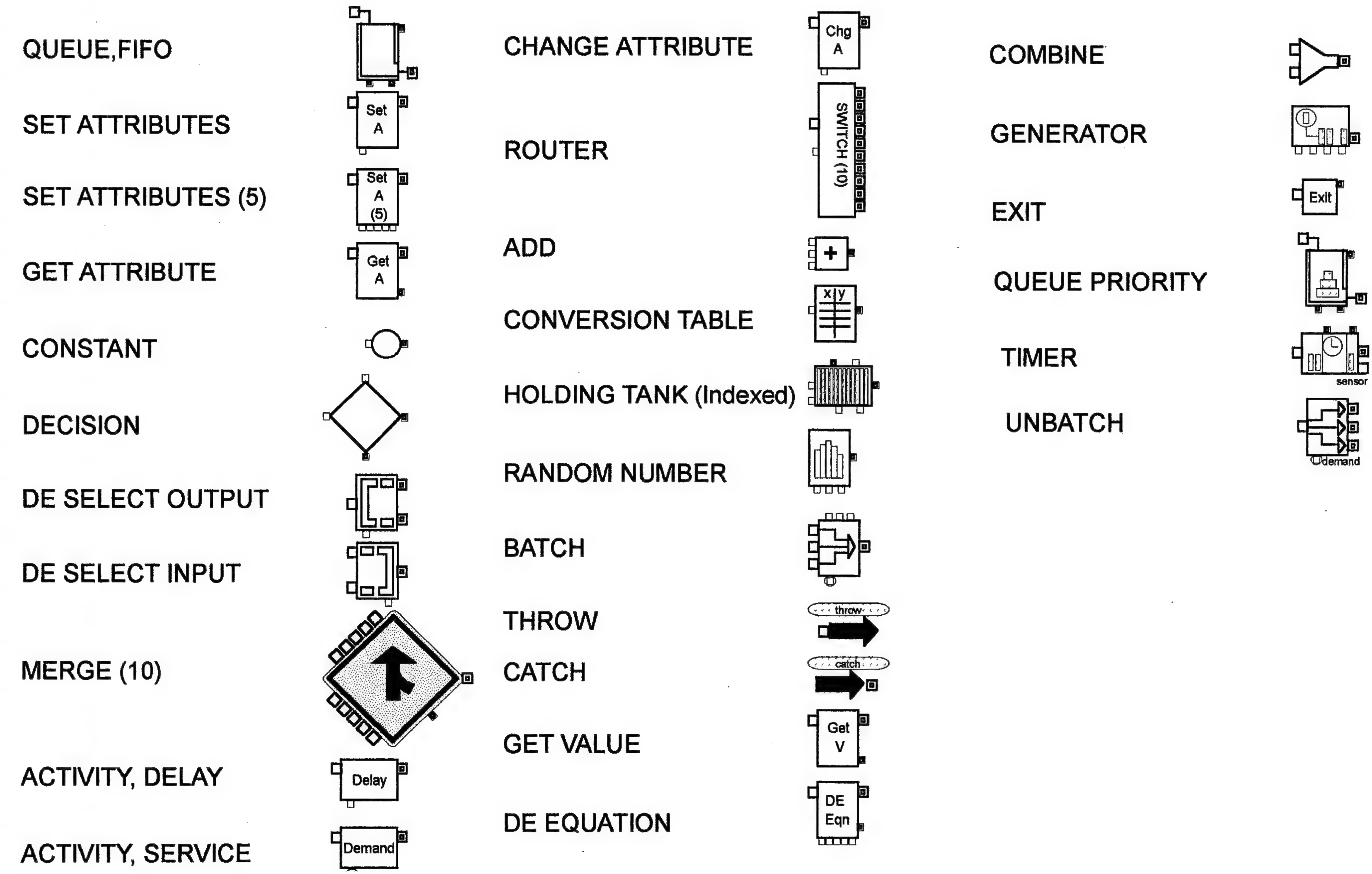
Partridge, Craig, *Gigabit Networking,* Addison-Wesley, 1994.

Understanding ATM Networking Applications, revision 2.0, American Research Group, 1996.

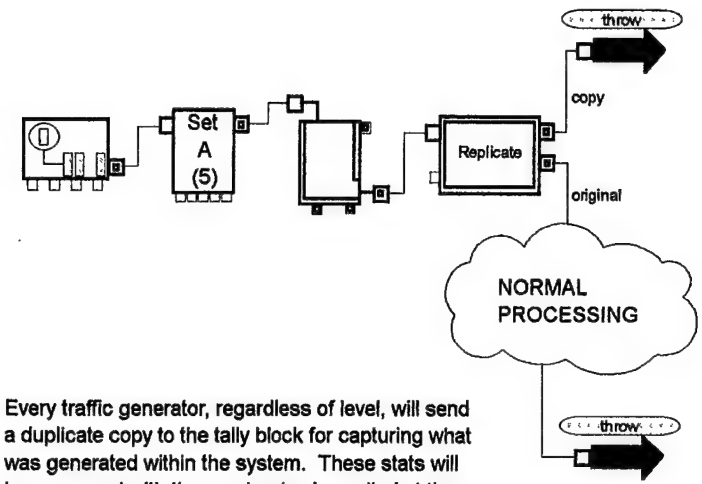
APPENDIX A. PAPER MODEL

The paper model was the result of the top-down decomposition of the complex BLII architecture system being modeled. This was the basis for the bottom-up design of the initial packet level model. The higher level decomposition remained unchanged through the redesign phase required by application performance.

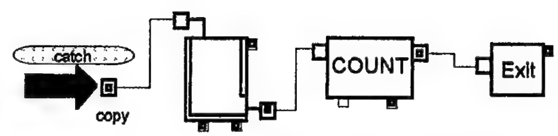
TEMPLATES


QUEUE,FIFO
SET ATTRIBUTES
Set A
SET ATTRIBUTES (5)
Set A (5)
GET ATTRIBUTE
Get A
CONSTANT
DECISION
DE SELECT OUTPUT
DE SELECT INPUT
MERGE (10)
ACTIVITY, DELAY
Delay
ACTIVITY, SERVICE
Demand
CHANGE ATTRIBUTE
Chg A
ROUTER
SWITCH (10)
ADD
+
CONVERSION TABLE
x y
HOLDING TANK (Indexed)
RANDOM NUMBER
BATCH
THROW
throw
CATCH
catch
GET VALUE
Get V
DE EQUATION
DE Eqn
COMBINE
GENERATOR
EXIT
Exit
QUEUE PRIORITY
TIMER
sensor
UNBATCH
demand

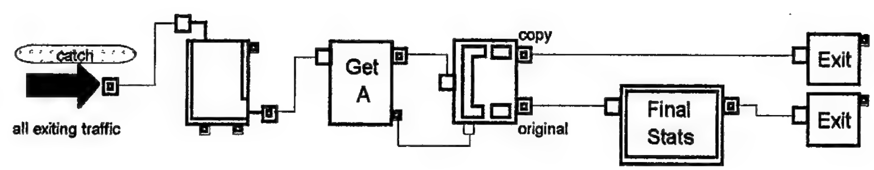
CAPTURING MEASURES OF EFFECTIVENESS (MOE)



Every traffic generator, regardless of level, will send a duplicate copy to the tally block for capturing what was generated within the system. These stats will be compared with the number having exited at the ending time of the run.

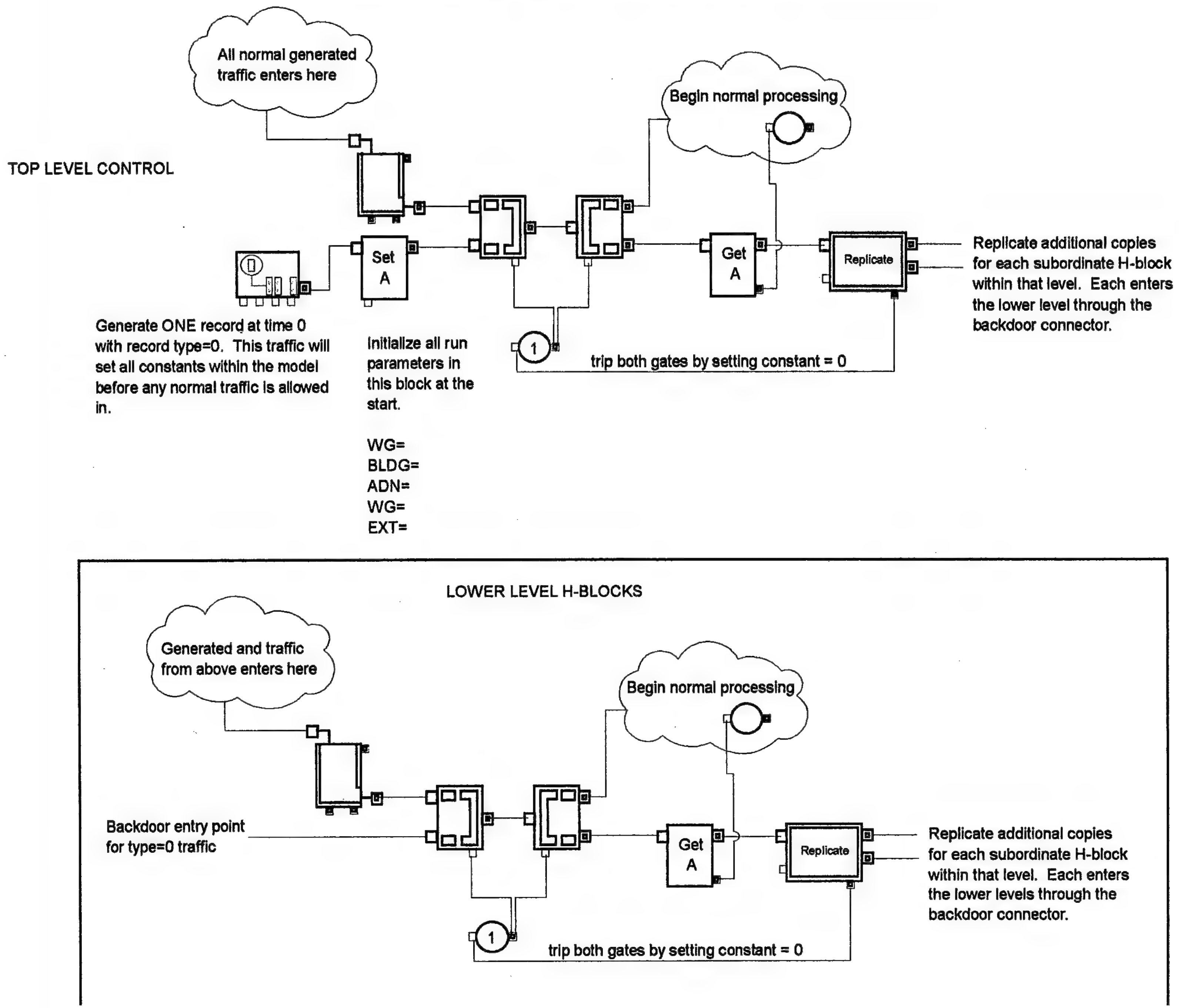


Count the number of traffic items generated within the system. May also gather other stats.



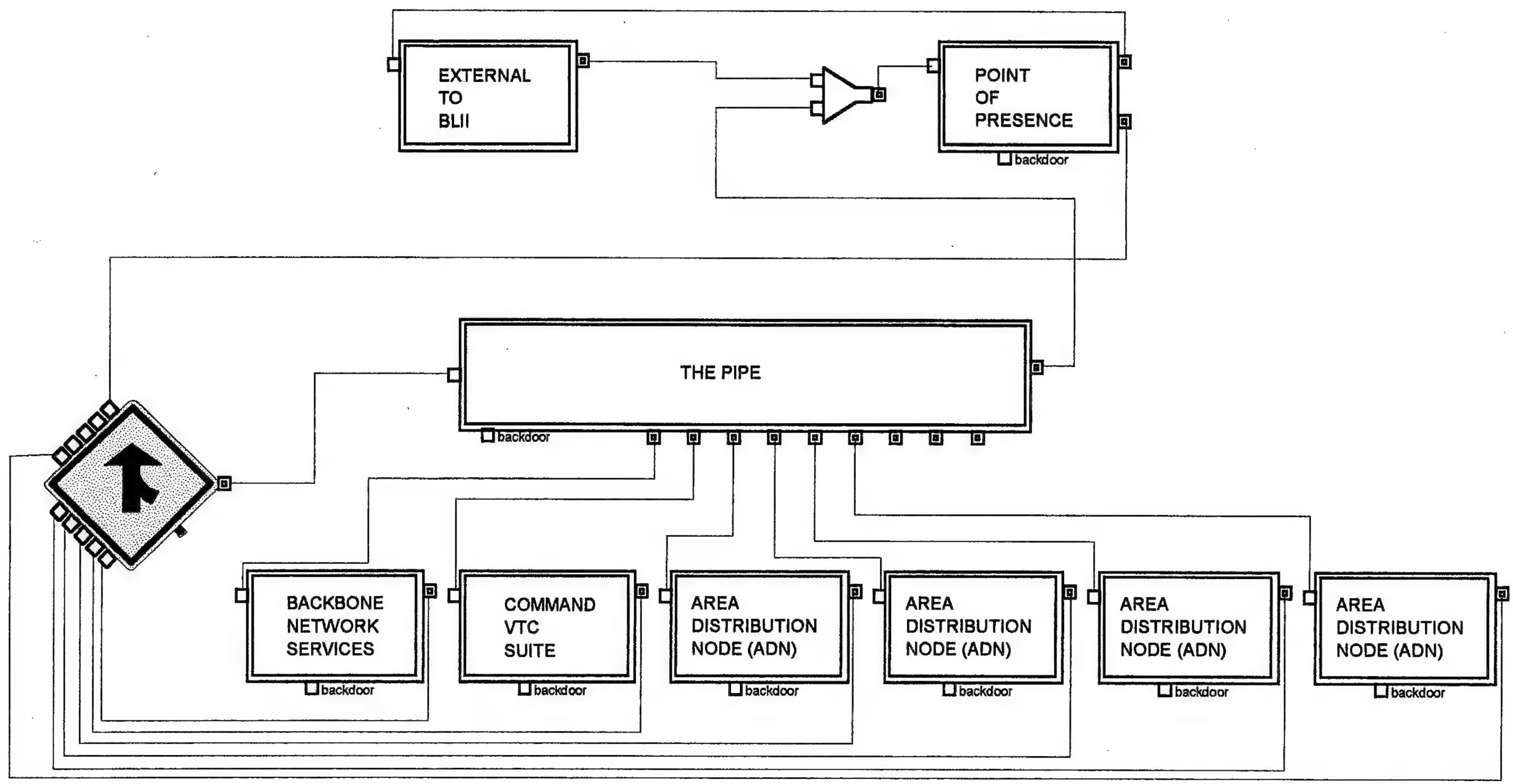
All traffic that passses through the network exits from this single point at the top level. Because some traffic may have multiple addressees, only the original copy will be kept for statistics. Duplicate copies of traffic will affect other traffic following behind in that particular pipe.

INITIALIZATION PROCESS

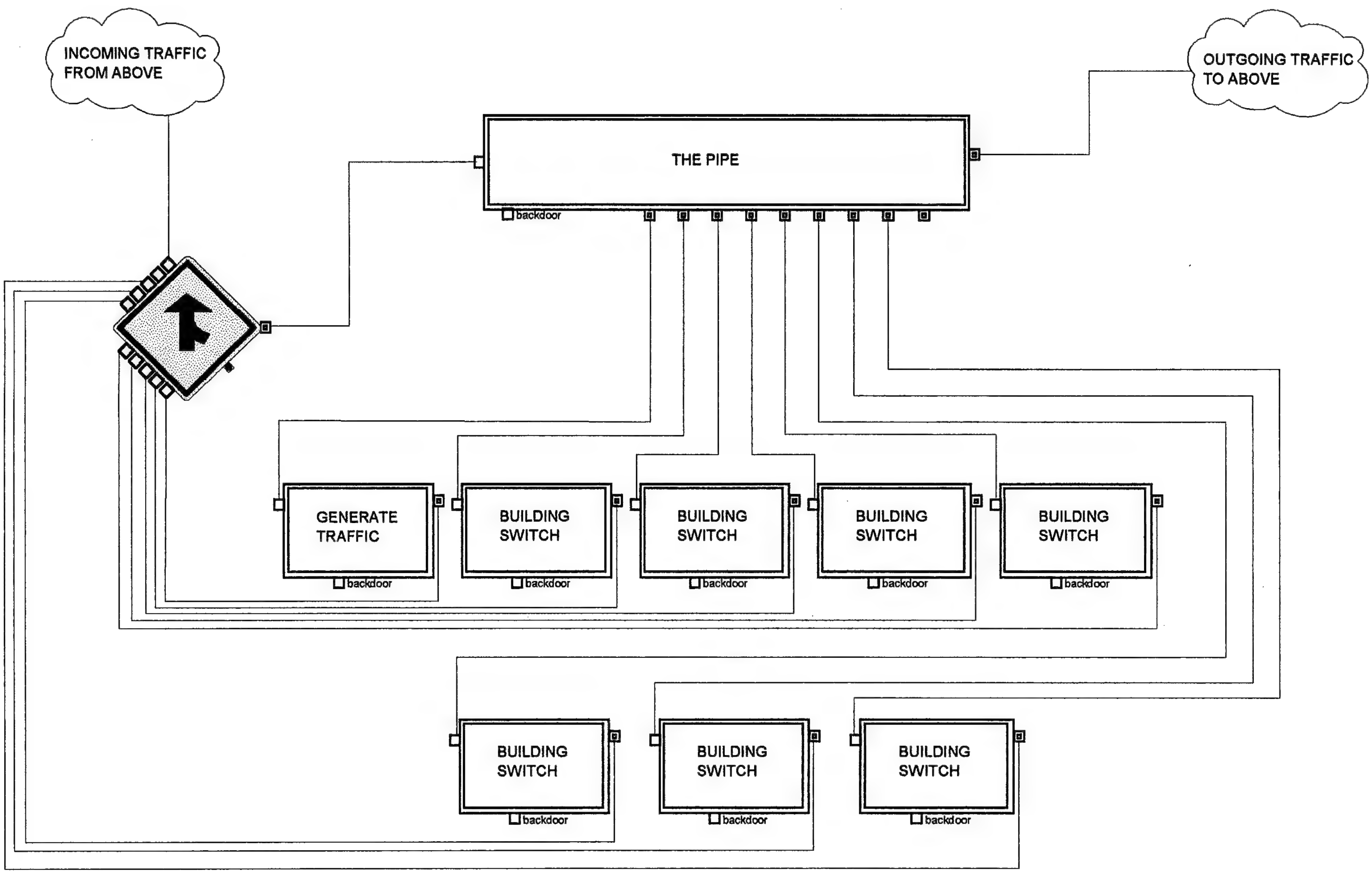


NOTE:
When the initialization records have proceeded through the lowest level of a branch, they are discarded and not treated as normal traffic.

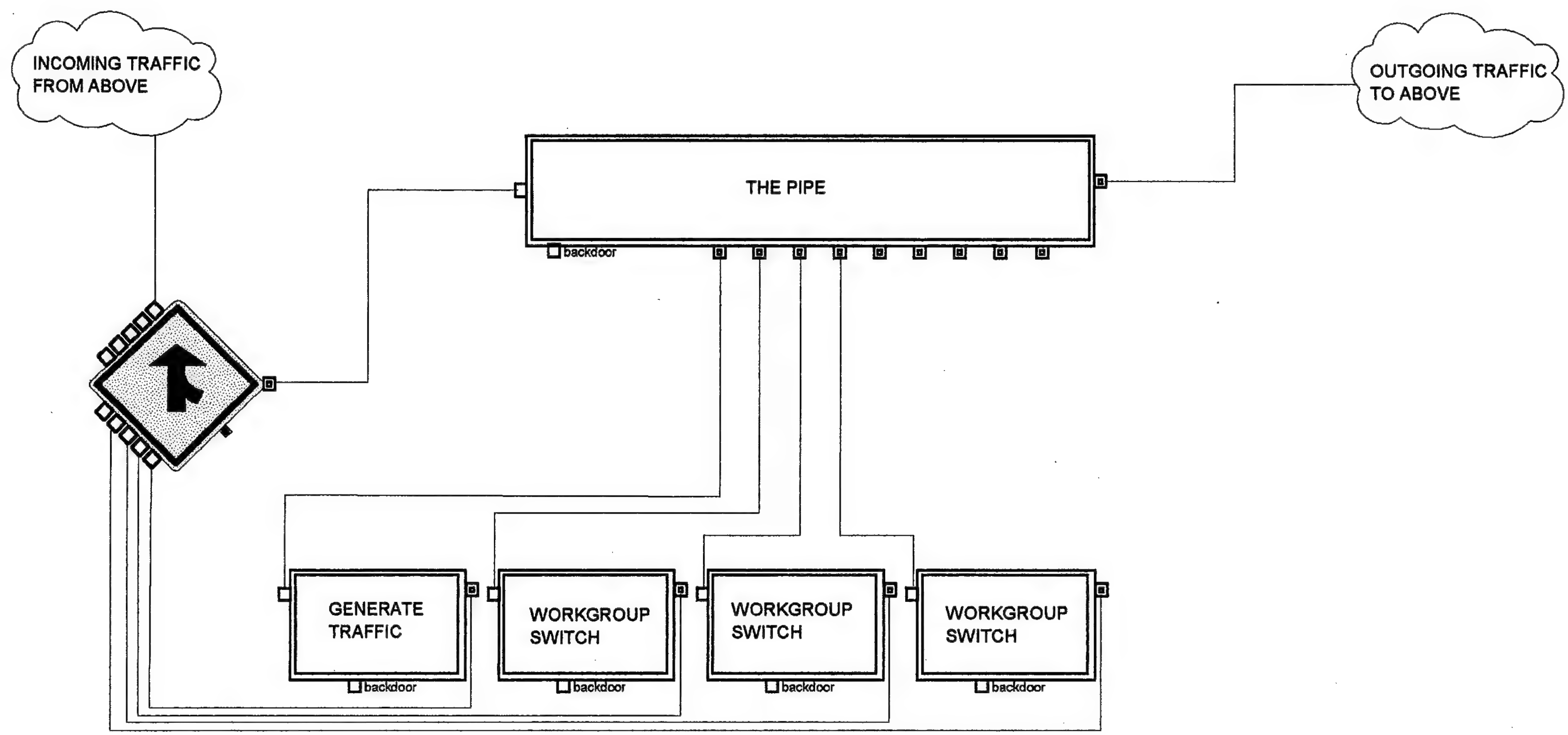
TOP LEVEL DIAGRAM



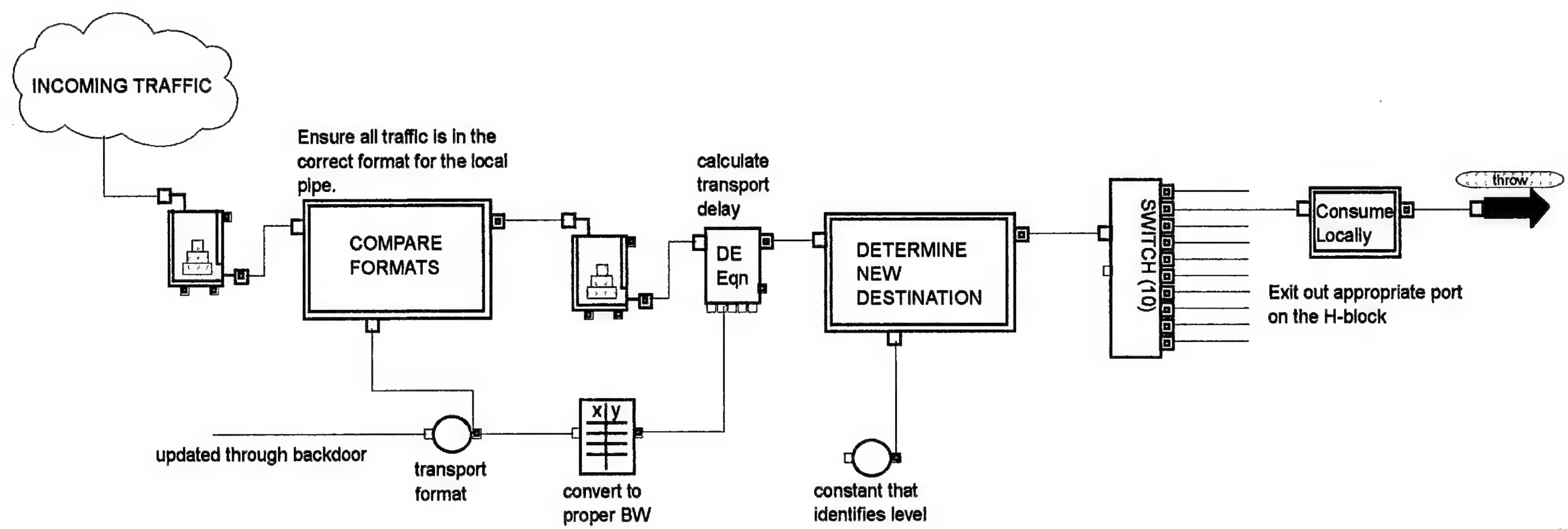
AREA DISTRIBUTION NODE DIAGRAM



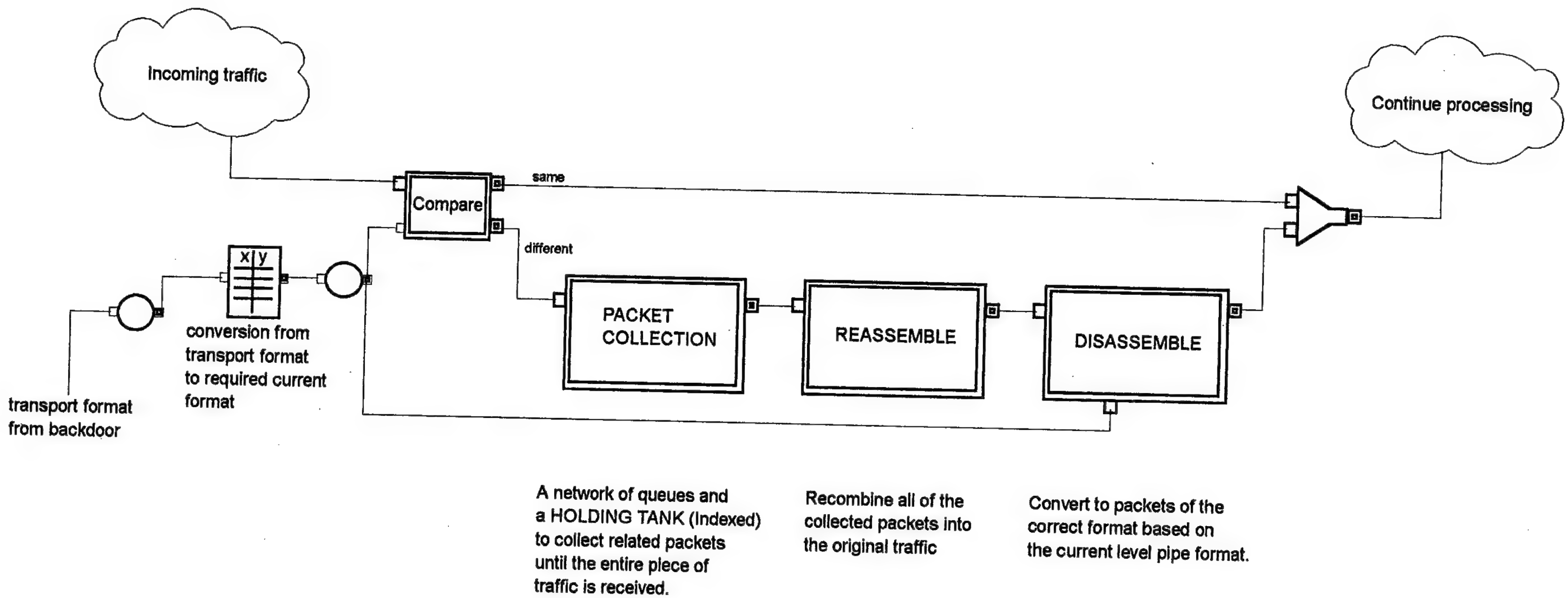
BUILDING SWITCH DIAGRAM



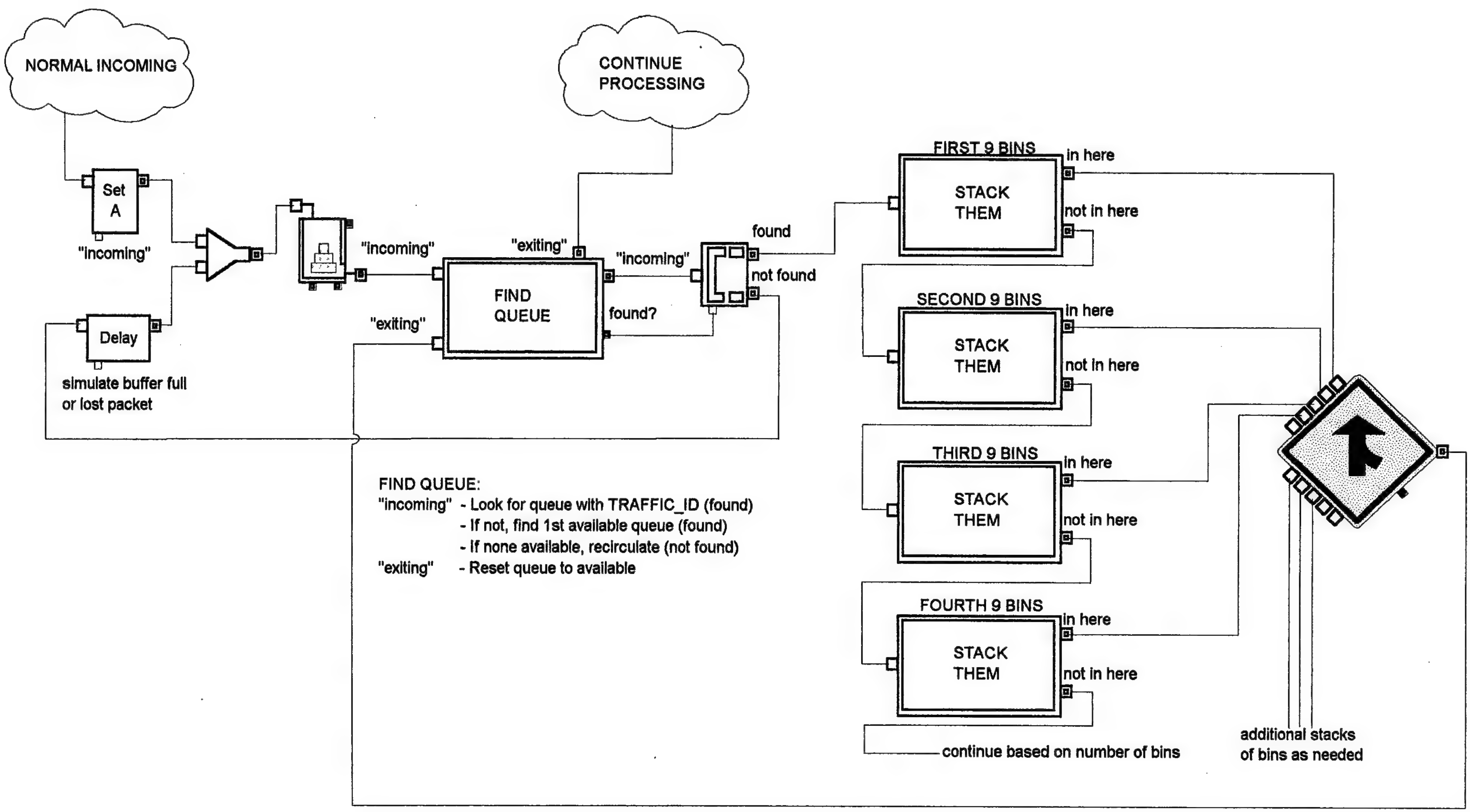
THE PIPE DIAGRAM



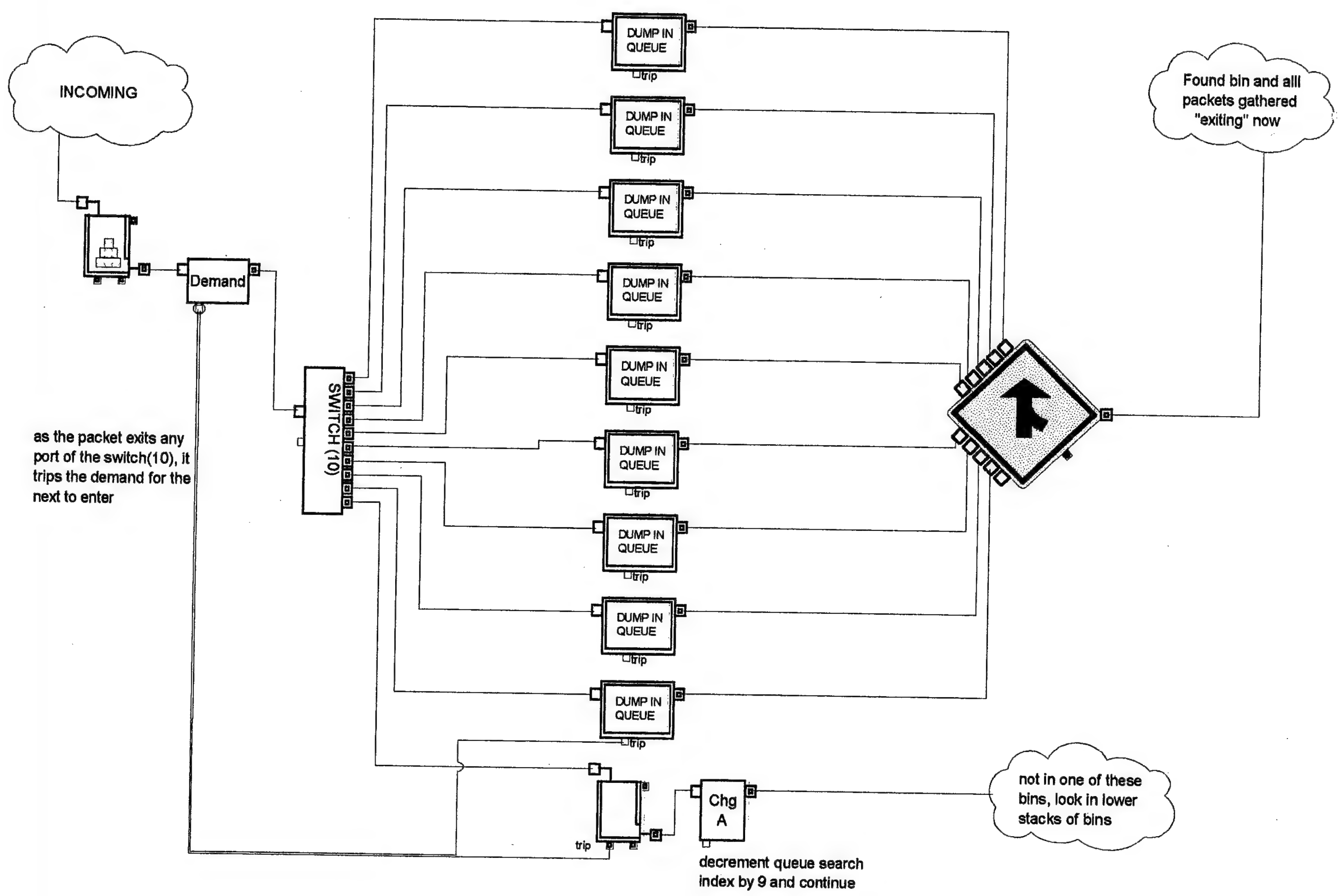
COMPARE FORMATS



PACKET COLLECTION

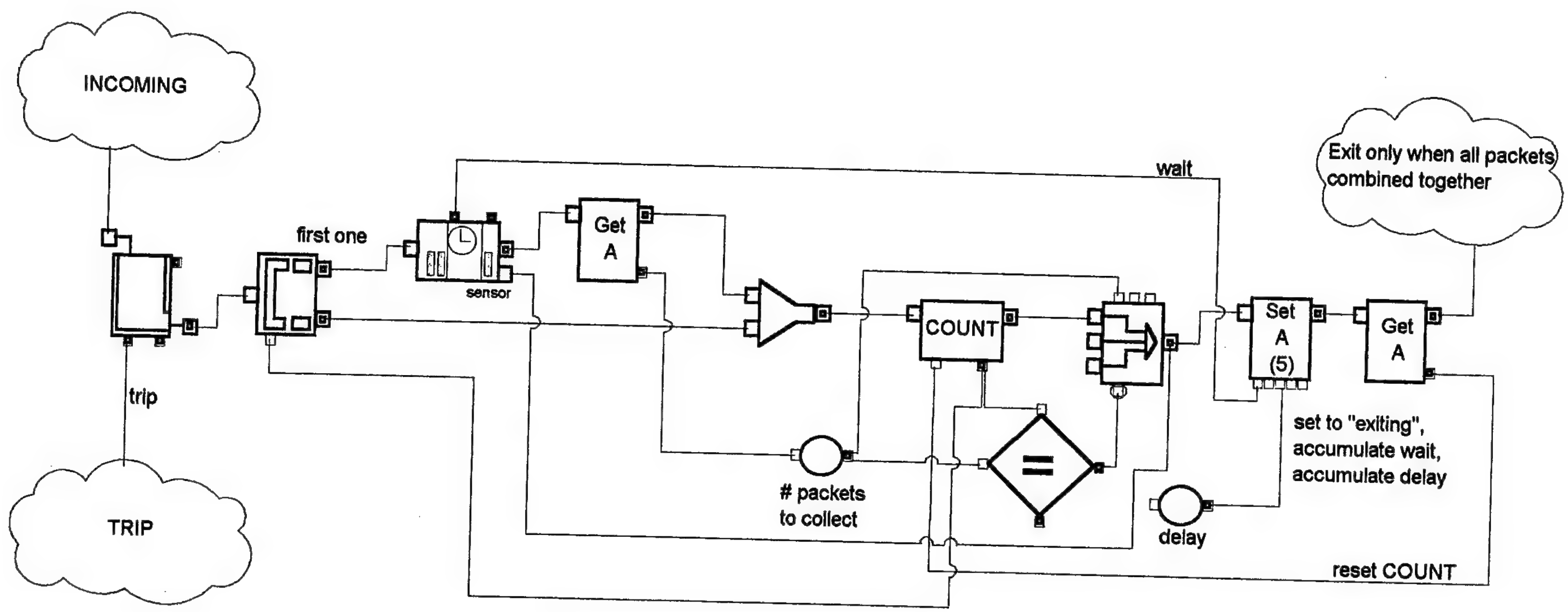


STACK THEM

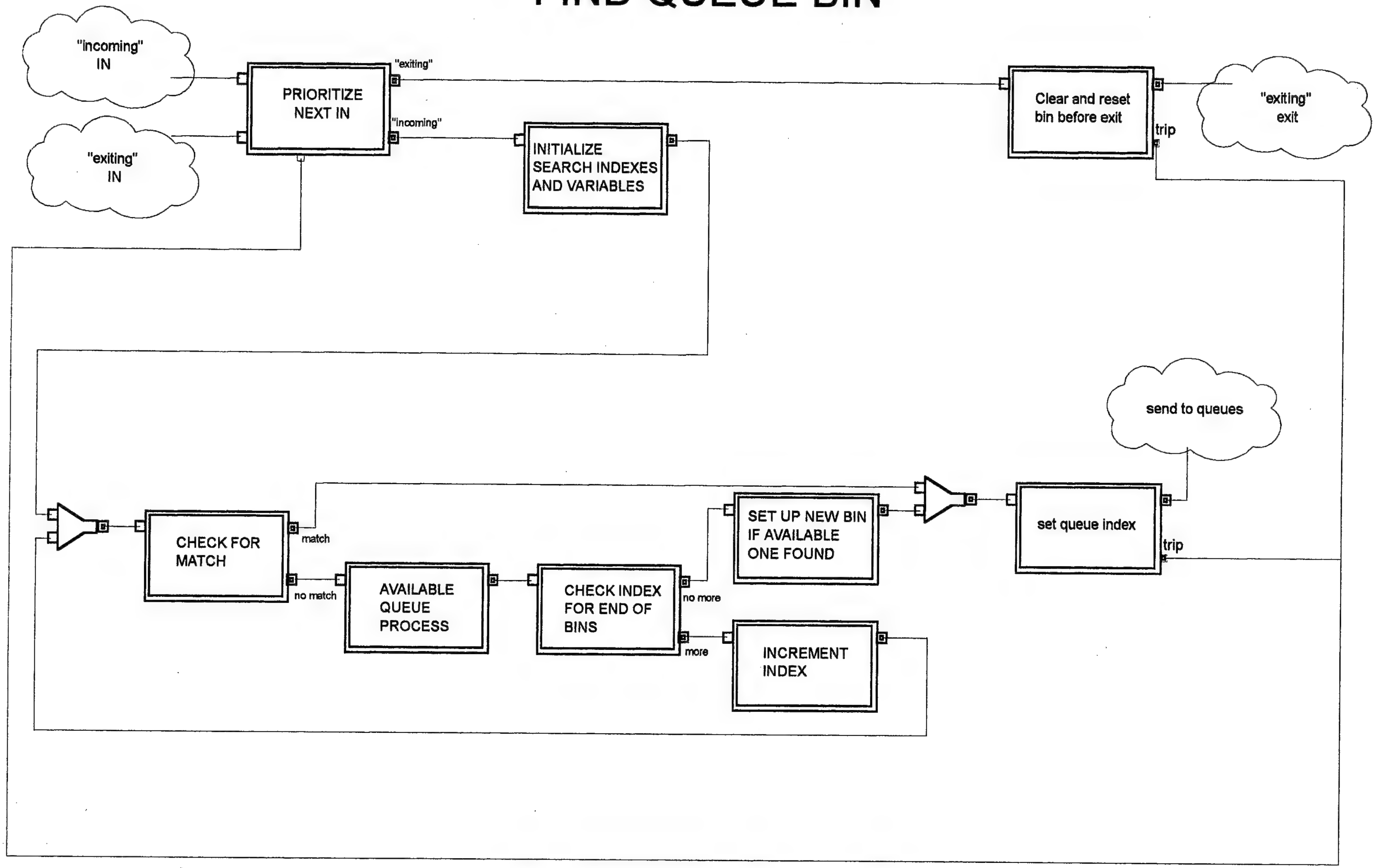


INCOMING
Demand
SWITCH (10)
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
DUMP IN QUEUE
trip
Found bin and alll packets gathered "exiting" now
as the packet exits any port of the switch(10), it trips the demand for the next to enter
trip
Chg A
not in one of these bins, look in lower stacks of bins
decrement queue search index by 9 and continue

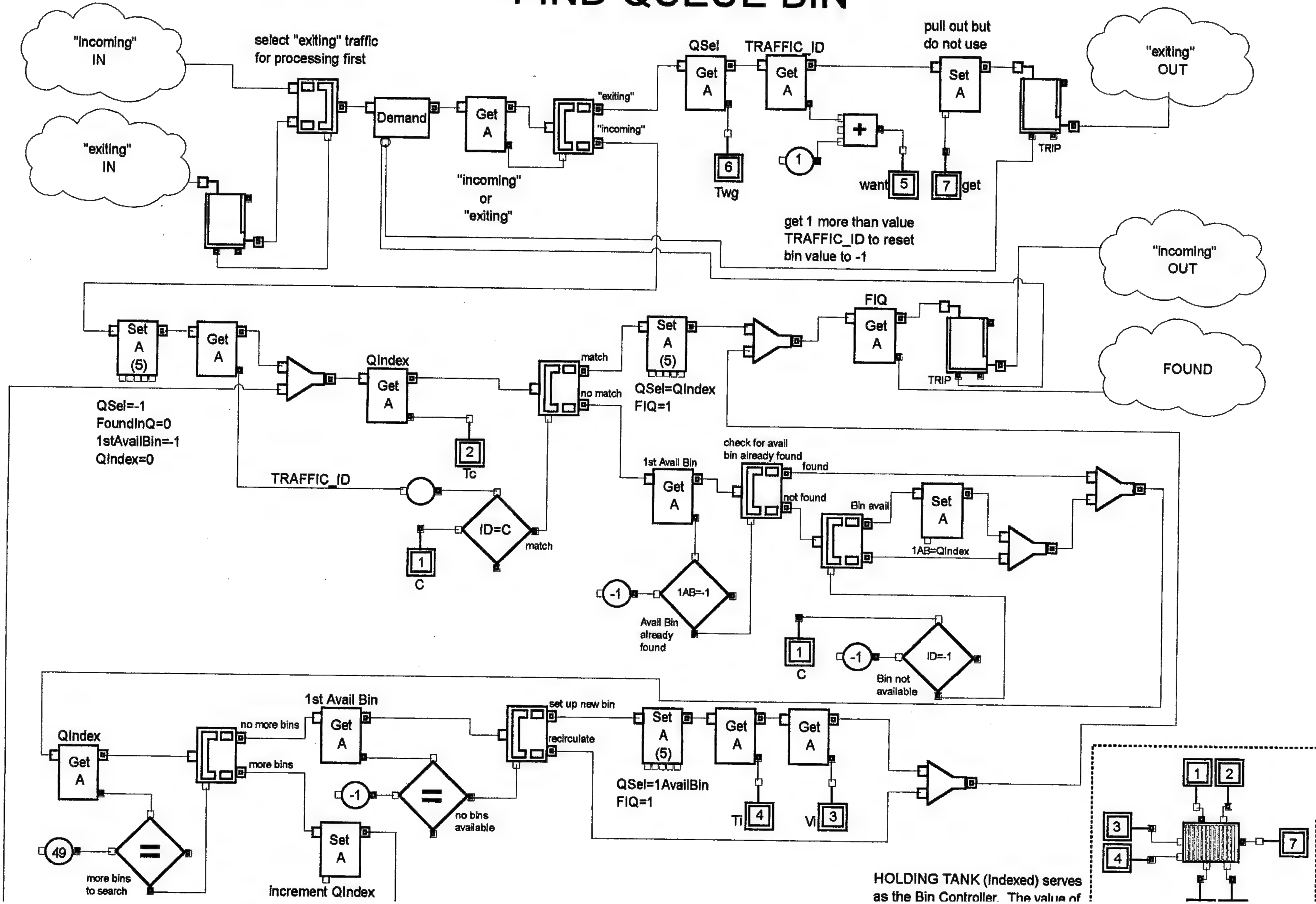
DUMP IN QUEUE



FIND QUEUE BIN

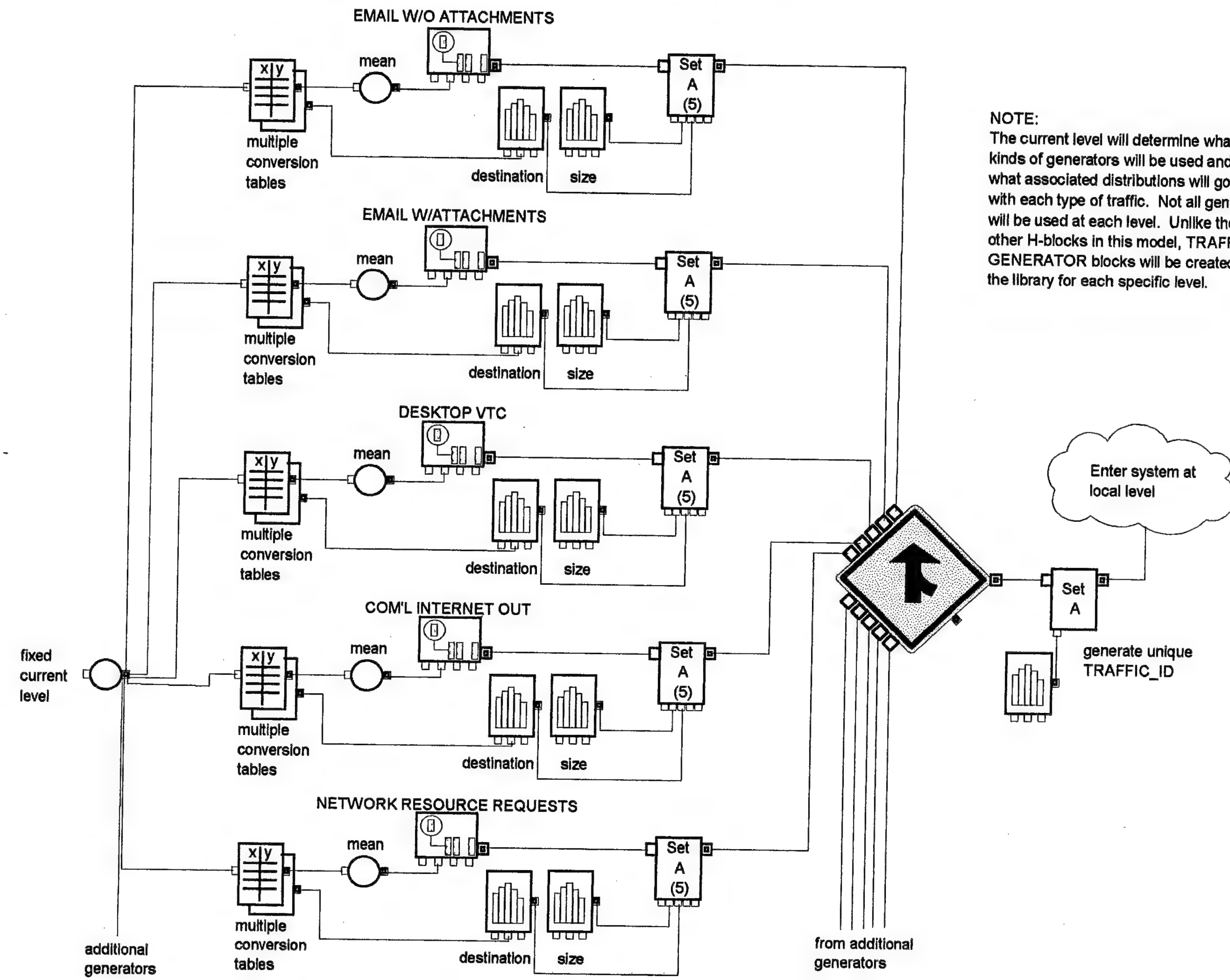


FIND QUEUE BIN



HOLDING TANK (Indexed) serves as the Bin Controller. The value of

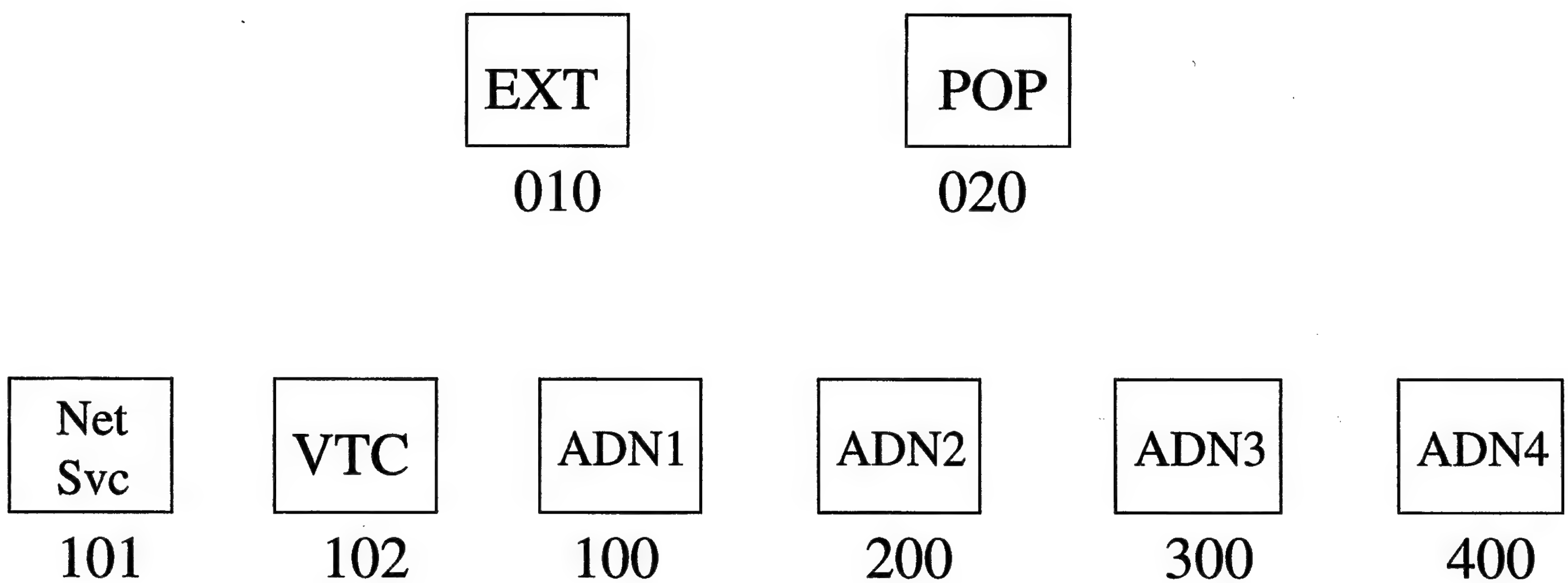
TRAFFIC GENERATION



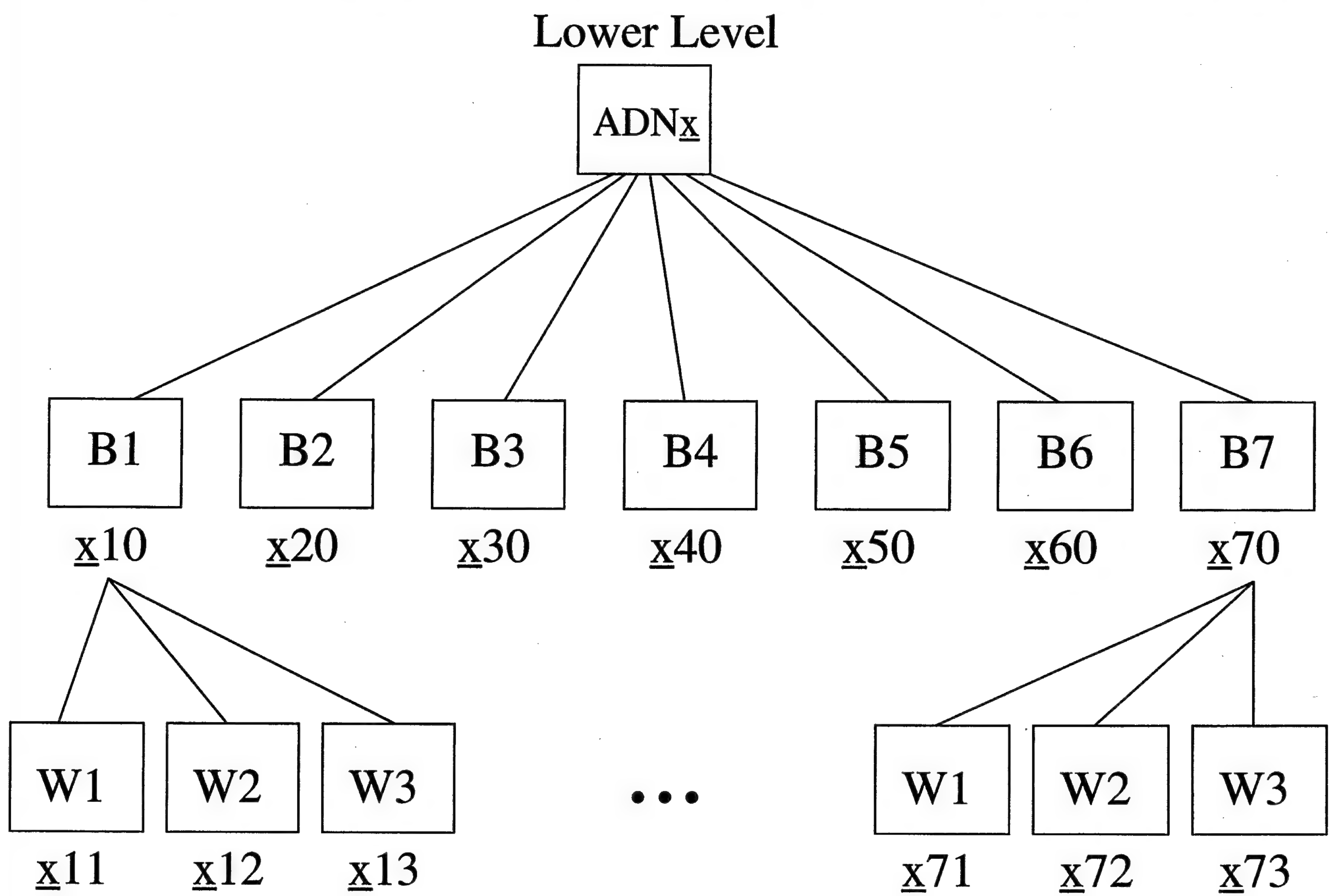
NOTE:
The current level will determine what kinds of generators will be used and what associated distributions will go with each type of traffic. Not all generators will be used at each level. Unlike the other H-blocks in this model, TRAFFIC GENERATOR blocks will be created in the library for each specific level.

SOURCE / DESTINATION ADDRESSING

Backbone Level



SOURCE / DESTINATION ADDRESSING



APPENDIX B. DETAILED MODEL RESULTS

This appendix is the spreadsheets containing the results of each model run. Each configuration was run three times with the results averaged for final analysis.

Normal Network Load												
		RUN 1			RUN 2			RUN 3			Average	
RUN #	Run Avg	# spikes > 0.05	Max Spike	Run Avg	# spikes > 0.05	Max Spike	Run Avg	# spikes > 0.05	Max Spike	Run Avg	# spikes > 0.05	Max Spike
01	0.0171	9	0.1959	0.0176	8	0.1721	0.0169	8	0.1888	0.0172	8.33	0.1856
02	0.0165	11	0.0962	0.0162	9	0.0901	0.0179	11	0.1101	0.0169	10.33	0.0988
03	0.0156	7	0.1173	0.0149	6	0.0922	0.0166	9	0.1821	0.0157	7.33	0.1305
04	2.4914	70	5.3321	1.8749	75	3.6825	3.7212	94	7.9597	2.6958	79.67	5.6581
05	3.9470	98	8.5156	1.4879	98	2.7384	1.4162	98	2.7771	2.2837	98.00	4.6770
06	0.1814	58	0.7384	0.4129	65	1.1040	0.2463	69	0.7657	0.2802	64.00	0.8694
07	0.0144	5	0.1069	0.0143	5	0.0989	0.0146	6	0.0997	0.0144	5.33	0.1018
08	0.0147	8	0.1121	0.0144	5	0.1014	0.0139	5	0.0856	0.0143	6.00	0.0997
09	0.0126	5	0.0811	0.0142	6	0.1005	0.0146	6	0.1111	0.0138	5.67	0.0976
10	0.0116	5	0.1275	0.0159	9	0.1224	0.0132	5	0.1559	0.0136	6.33	0.1353
11	0.0127	7	0.0739	0.0130	6	0.0601	0.0139	6	0.0855	0.0132	6.33	0.0732
12	0.0121	8	0.0816	0.0143	6	0.0852	0.0111	7	0.0812	0.0125	7.00	0.0827
13	0.0137	7	0.0872	0.0122	4	0.0666	0.0135	6	0.0854	0.0131	5.67	0.0797
14	0.0153	9	0.1131	0.0149	8	0.1251	0.0156	12	0.1023	0.0153	9.67	0.1135
Medium Workload Increase												
01	0.0238	8	0.1003	0.0255	9	0.1101	0.0209	8	0.1043	0.0234	8.33	0.1049
02	0.0249	11	0.1050	0.0243	8	0.1118	0.0205	9	0.0964	0.0232	9.33	0.1044
03	0.0201	8	0.0920	0.0217	12	0.1117	0.0182	7	0.1117	0.0200	9.00	0.1051
07	0.0203	7	0.0773	0.0104	2	0.0517	0.0146	4	0.0738	0.0151	4.33	0.0676
08	0.0155	6	0.0701	0.0198	6	0.0850	0.0141	4	0.0699	0.0165	5.33	0.0750
09	0.0124	2	0.0560	0.0150	5	0.0734	0.0199	9	0.0796	0.0158	5.33	0.0697
12	0.0180	0	0.0498	0.0143	0	0.0498	0.0123	4	0.0809	0.0149	1.33	0.0602
Heavy Workload Increase												
02	0.0523	50	0.1281	0.0388	25	0.1447	0.0365	19	0.0988	0.0425	31.33	0.1239
03	0.0327	15	0.0773	0.0394	15	0.0788	0.0387	14	0.2465	0.0369	14.67	0.1342
07	0.0499	48	0.1056	0.0268	11	0.0682	0.0318	10	0.0688	0.0362	23.00	0.0809
09	0.0407	30	0.1221	0.0334	16	0.0882	0.0455	39	0.1247	0.0399	28.33	0.1117
12	0.0427	24	0.0968	0.0429	30	0.1067	0.0295	3	0.0766	0.0384	19.00	0.0934

Normal Workload

RUN #	External	Backbone	ADN	Building	Workgroup	Run Avg	# spikes > 0.05	Max Spike
01	Classic IP(4)	F-E/N (6)	F-E/N (6)	F-E/N (6)	E/N (7)	0.0172	8.33	0.1856
02	Classic IP(4)	Gb E/N (5)	F-E/N (6)	F-E/N (6)	E/N (7)	0.0169	10.33	0.0988
03	Classic IP(4)	Gb E/N (5)	Gb E/N (5)	F-E/N (6)	E/N (7)	0.0157	7.33	0.1305
04	Classic IP(4)	FDDI (0)	F-E/N (6)	F-E/N (6)	E/N (7)	2.6958	79.67	5.6581
05	Classic IP(4)	FDDI (0)	Gb E/N (5)	F-E/N (6)	E/N (7)	2.2837	98.00	4.6770
06	Classic IP(4)	FDDI (0)	FDDI (0)	F-E/N (6)	E/N (7)	0.2802	64.00	0.8694
07	Classic IP(4)	ATM OC-12 (1)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0144	5.33	0.1018
08	Classic IP(4)	ATM OC-12 (1)	ATM OC-12 (1)	F-E/N (6)	E/N (7)	0.0143	6.00	0.0997
09	Classic IP(4)	ATM OC-3 (2)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0138	5.67	0.0976
10	Classic IP(4)	ATM OC-12 (1)	ATM OC-3 (2)	ATM OC-3 (2)	E/N (7)	0.0136	6.33	0.1353
11	Classic IP(4)	ATM OC-12 (1)	ATM OC-3 (2)	ATM OC-3 (2)	ATM DS-3 (3)	0.0132	6.33	0.0732
12	Classic IP(4)	ATM OC-12 (1)	ATM OC-12 (1)	ATM OC-3 (2)	ATM DS-3 (3)	0.0125	7.00	0.0827
13	ATM OC-3 (2)	ATM OC-12 (1)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0131	5.67	0.0797
14	Classic IP(4)	Classic IP(4)	Classic IP(4)	Classic IP(4)	E/N (7)	0.0153	9.67	0.1135

Medium Workload Increase

01	Classic IP(4)	F-E/N (6)	F-E/N (6)	F-E/N (6)	E/N (7)	0.0234	8.33	0.1049
02	Classic IP(4)	Gb E/N (5)	F-E/N (6)	F-E/N (6)	E/N (7)	0.0232	9.33	0.1044
03	Classic IP(4)	Gb E/N (5)	Gb E/N (5)	F-E/N (6)	E/N (7)	0.0200	9.00	0.1051
07	Classic IP(4)	ATM OC-12 (1)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0151	4.33	0.0676
08	Classic IP(4)	ATM OC-12 (1)	ATM OC-12 (1)	F-E/N (6)	E/N (7)	0.0165	5.33	0.0750
09	Classic IP(4)	ATM OC-3 (2)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0158	5.33	0.0697
12	Classic IP(4)	ATM OC-12 (1)	ATM OC-12 (1)	ATM OC-3 (2)	ATM DS-3 (3)	0.0149	1.33	0.0602

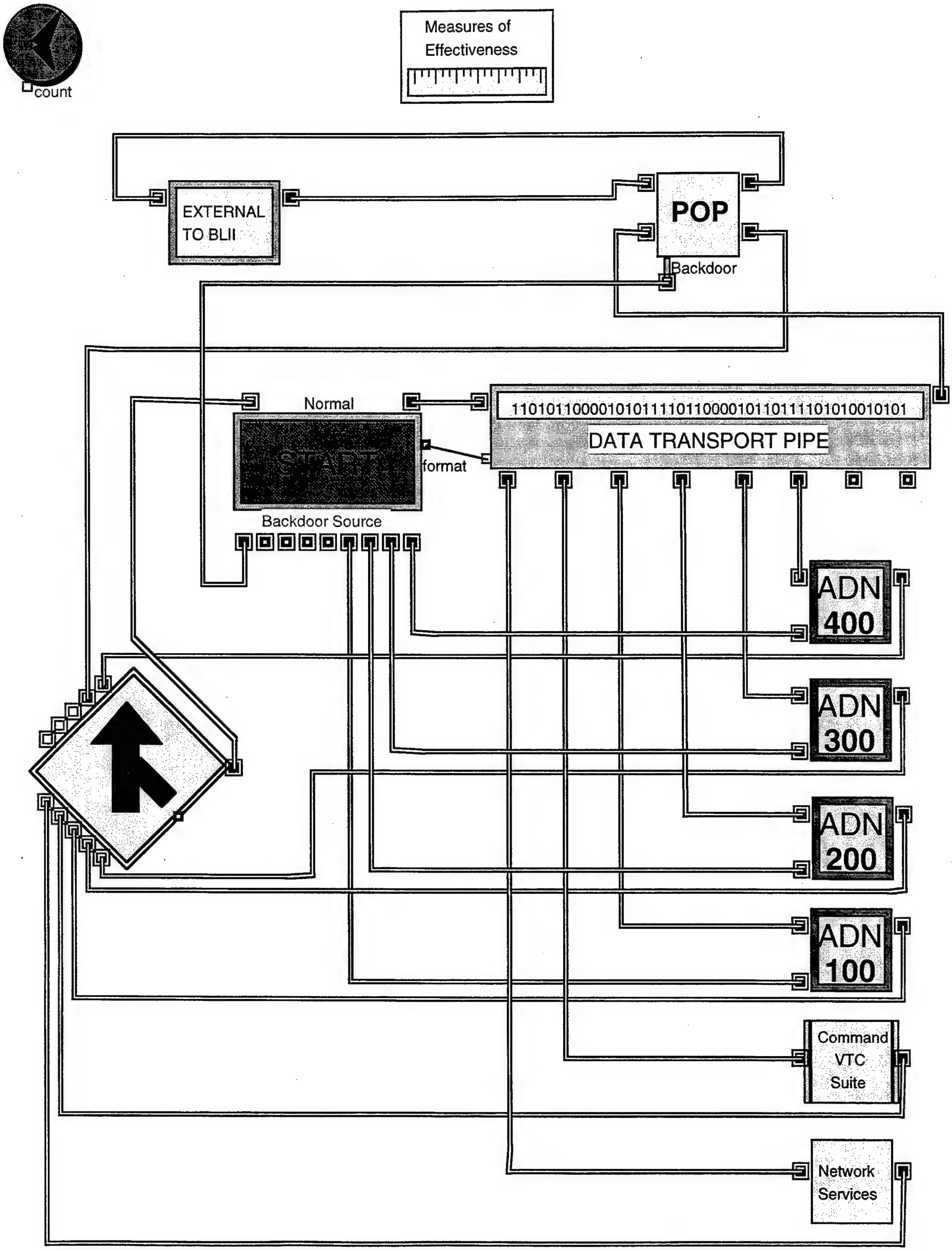
Heavy Workload Increase

02	Classic IP(4)	Gb E/N (5)	F-E/N (6)	F-E/N (6)	E/N (7)	0.0425	31.33	0.1239
03	Classic IP(4)	Gb E/N (5)	Gb E/N (5)	F-E/N (6)	E/N (7)	0.0369	14.67	0.1342
07	Classic IP(4)	ATM OC-12 (1)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0362	23.00	0.0809
09	Classic IP(4)	ATM OC-3 (2)	ATM OC-3 (2)	F-E/N (6)	E/N (7)	0.0399	28.33	0.1117
12	Classic IP(4)	ATM OC-12 (1)	ATM OC-12 (1)	ATM OC-3 (2)	ATM DS-3 (3)	0.0384	19.00	0.0934

APPENDIX C. EXTEND BLII ARCHITECTURE BLOCKS

The Extend blocks in this appendix are the high level architecture blocks modeling the physical infrastructure of the BLII. The following hierarchical blocks are contained in this appendix:

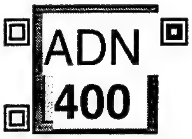
- BLII Model
- Area Distribution Node block
- Building block
- Workgroup block
- External block
- Point of Presence block



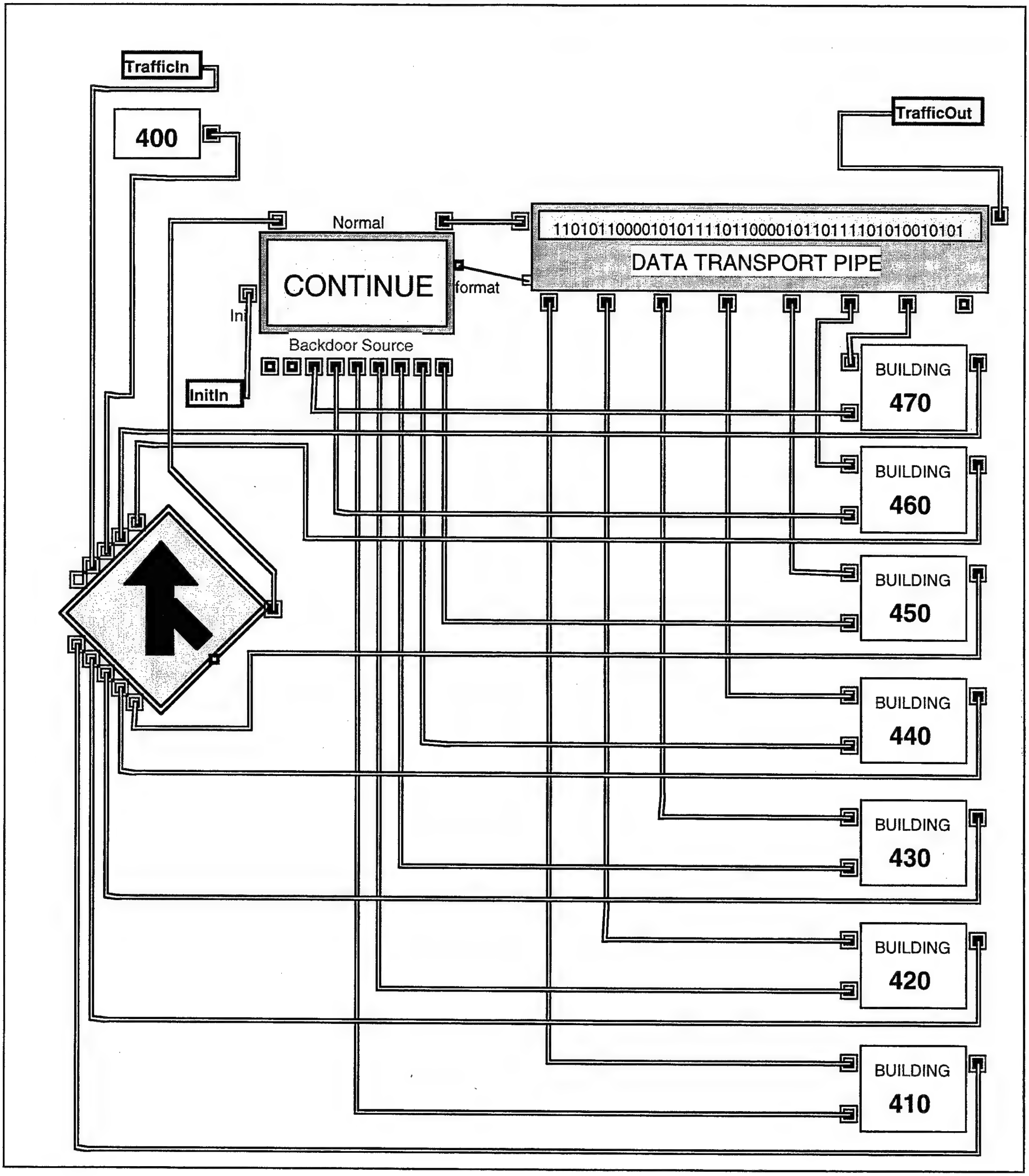
count
Measures of
Effectiveness
EXTERNAL
TO BLII
POP
Backdoor
Normal
110101100001010111011000010110111101010010101
DATA TRANSPORT PIPE
format
Backdoor Source
ADN
400
ADN
300
ADN
200
ADN
100
Command
VTC
Suite
Network
Services

Structure of ADN

Icon of block ADN



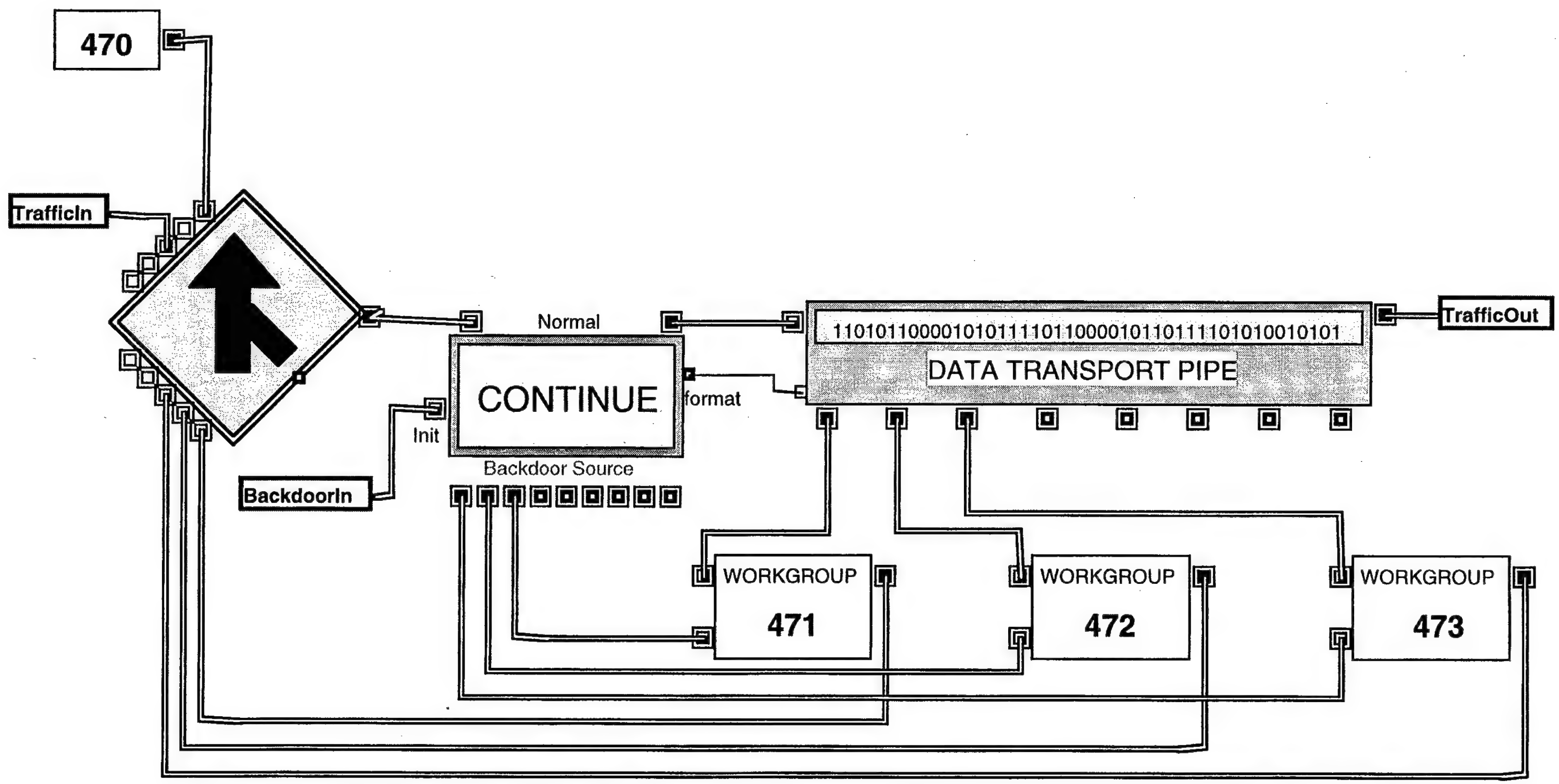
Structure of ADN



Structure of Building

Icon of block Building

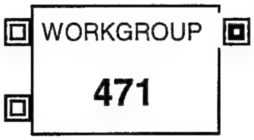
Structure of Building



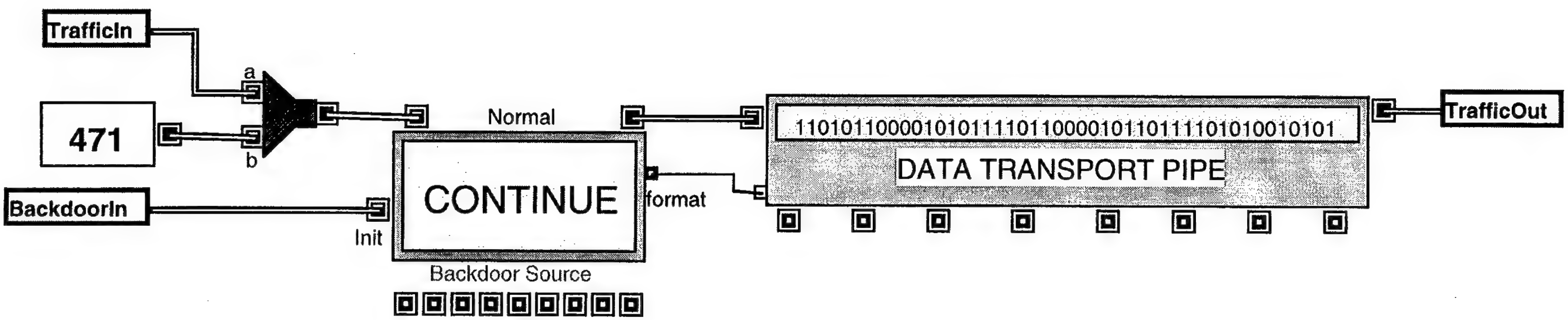
Building - 1

Structure of Workgroup

Icon of block Workgroup



Structure of Workgroup

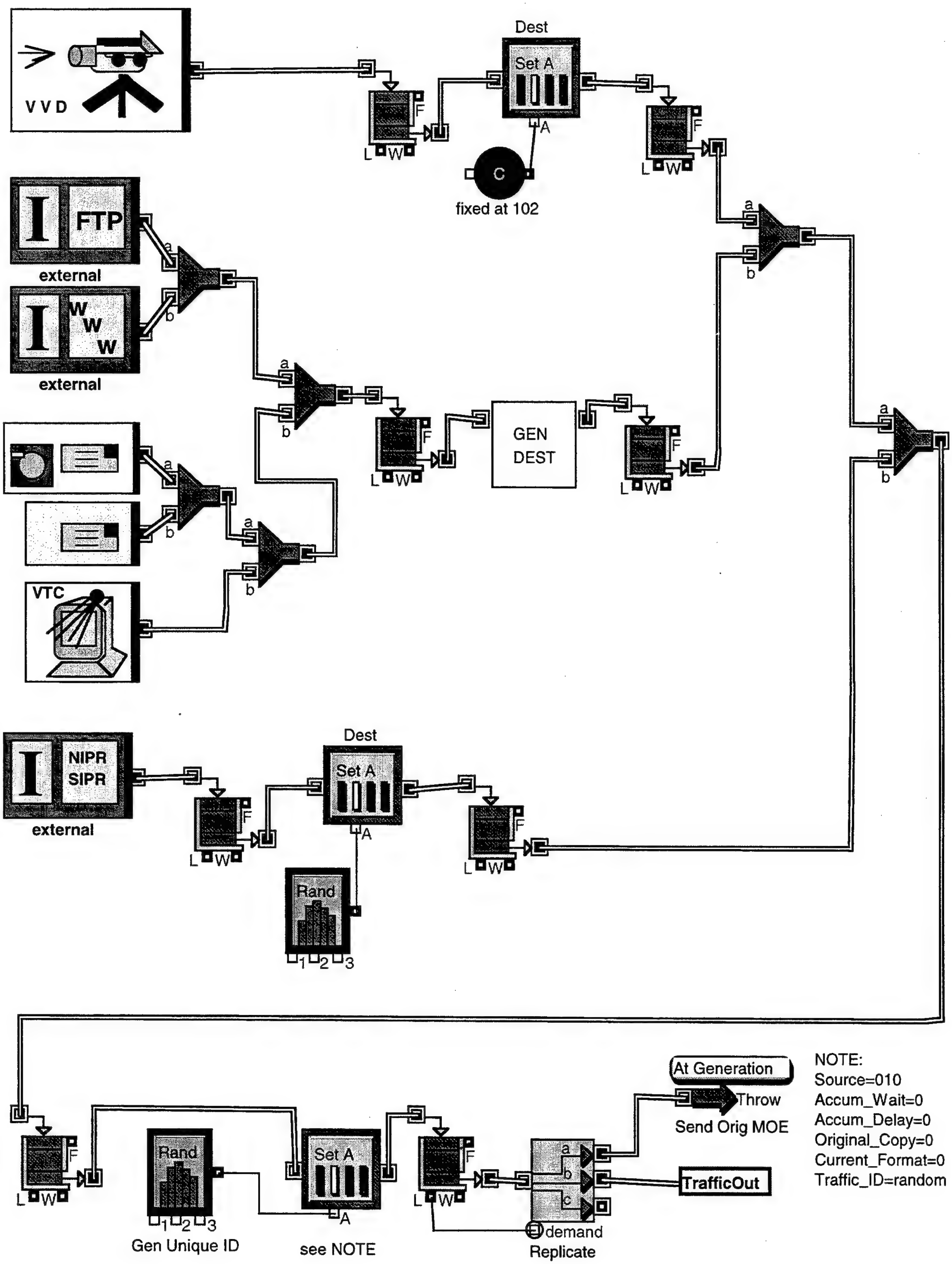


Workgroup - 1

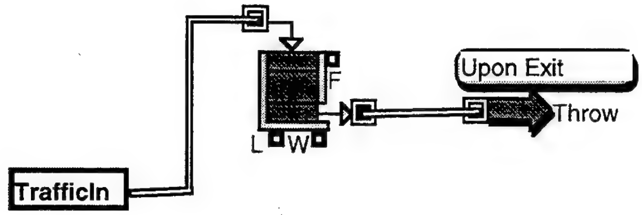
Structure of External Traffic

Icon of block External Traffic

Structure of External Traffic



Structure of External Traffic

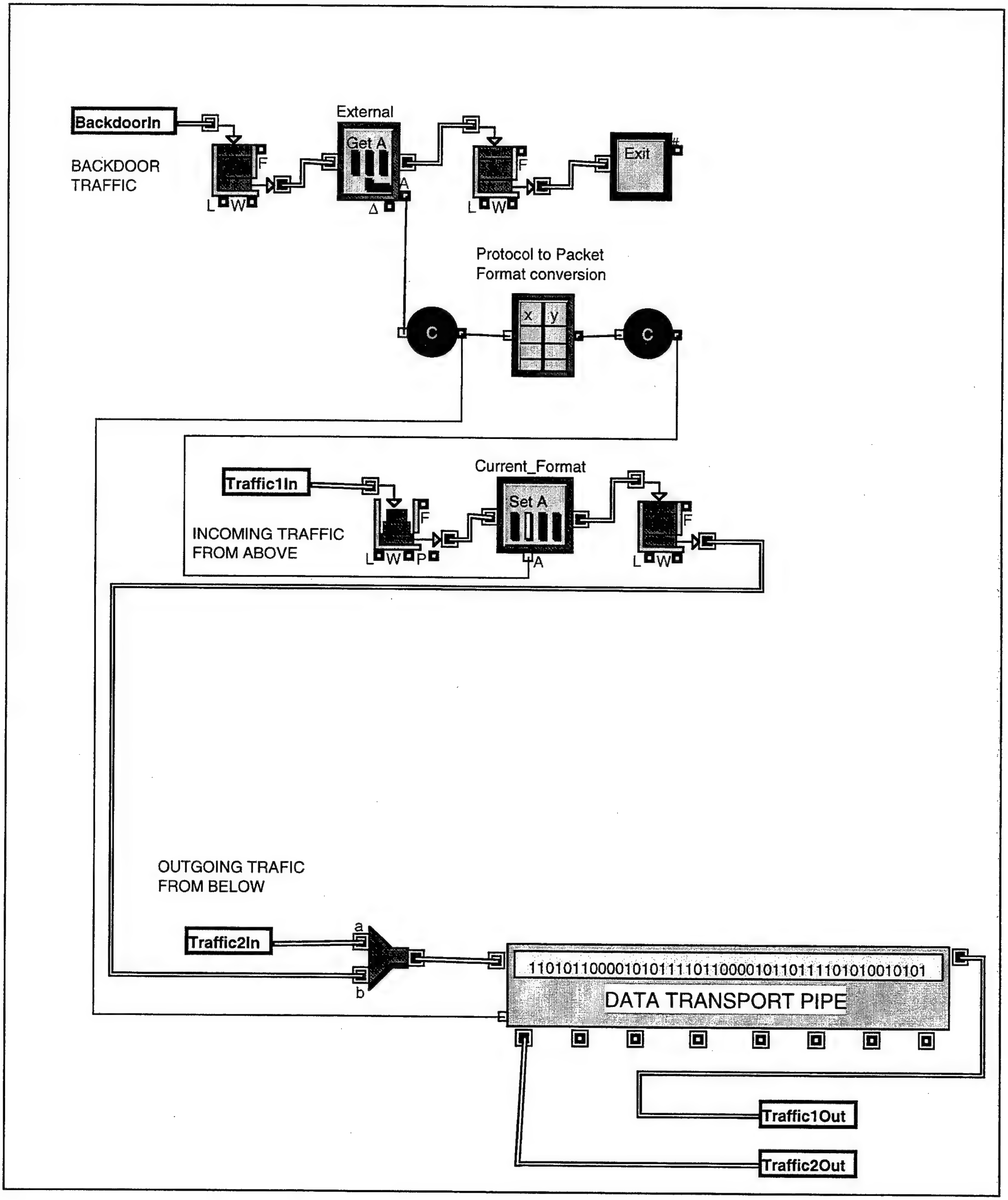


AT THIS TIME, NO FURTHER PROCESSING OF INCOMING TRAFFIC IS BEING CONDUCTED.

Structure of Point of Presence

Icon of block Point of Presence

Structure of Point of Presence



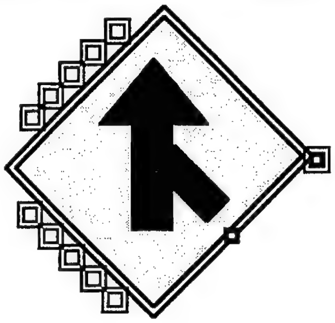
APPENDIX D. EXTEND GENERAL UTILITY BLOCKS

The Extend blocks in this appendix are the utility blocks used at various locations throughout the infrastructure of the BLII. These blocks do not perform a specific function relating to the processing of network traffic, but rather assist in implementing the implementation of the model specifically in the Extend application environment. The following hierarchical blocks are contained in this appendix:

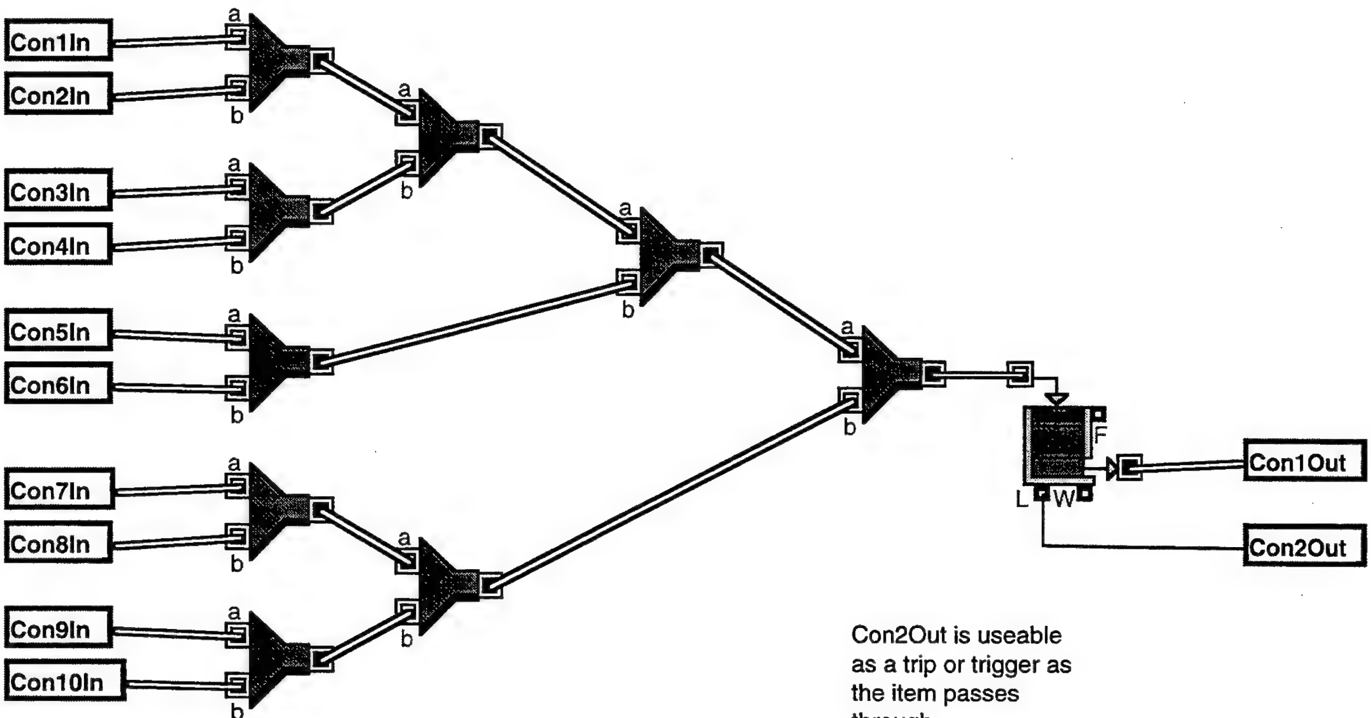
- Merge10
- Select Output Path (10)
- Regenerate 9
- Start Initialization
- Continue Initialization
- Measures of Effectiveness

Structure of Merge10 (ATMNETWORK.LIX)

Icon of block Merge10

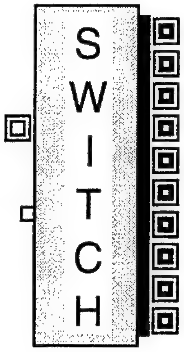


Structure of Merge10 (ATMNETWORK.LIX)

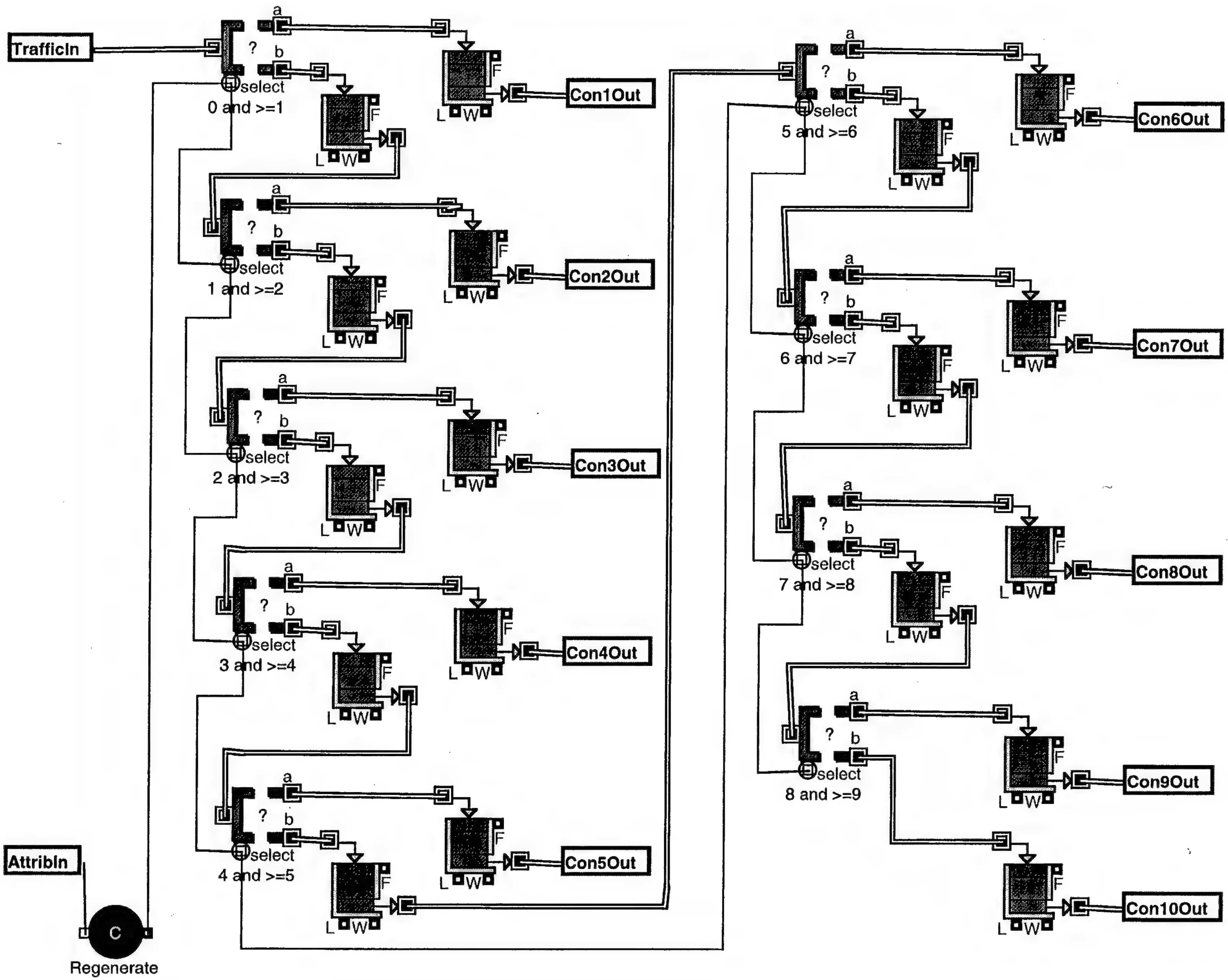


Con2Out is useable as a trip or trigger as the item passes through

Icon of block Select Output Path (10)

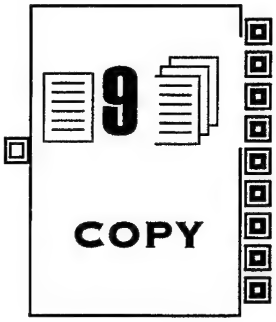


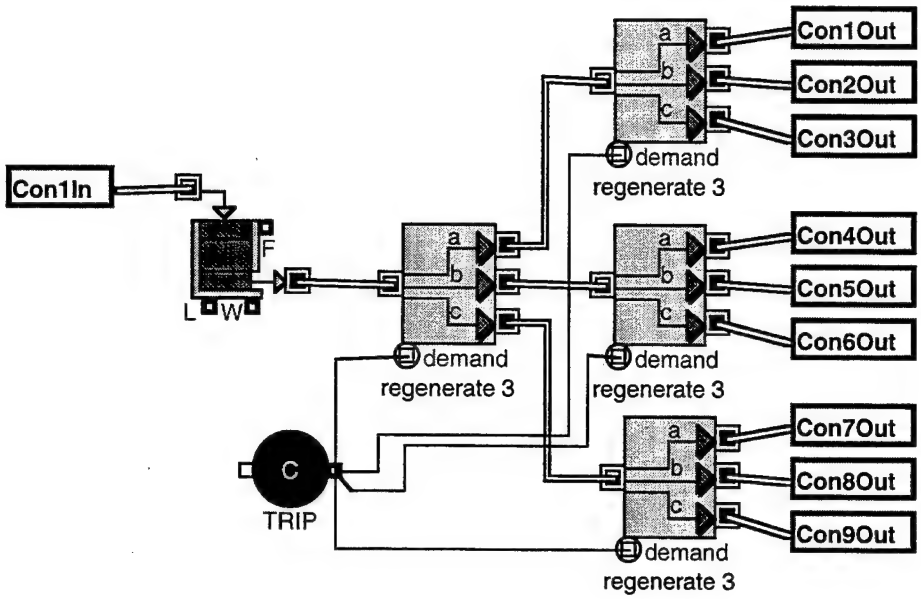
Structure of Select Output Path (10) (ATMNETWORK.LIX)



Select Output Path (10) - 1

Icon of block Regenerate 9

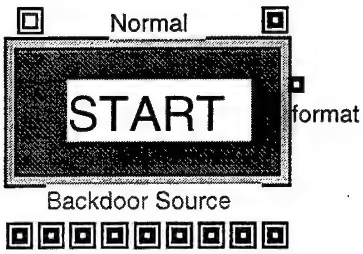




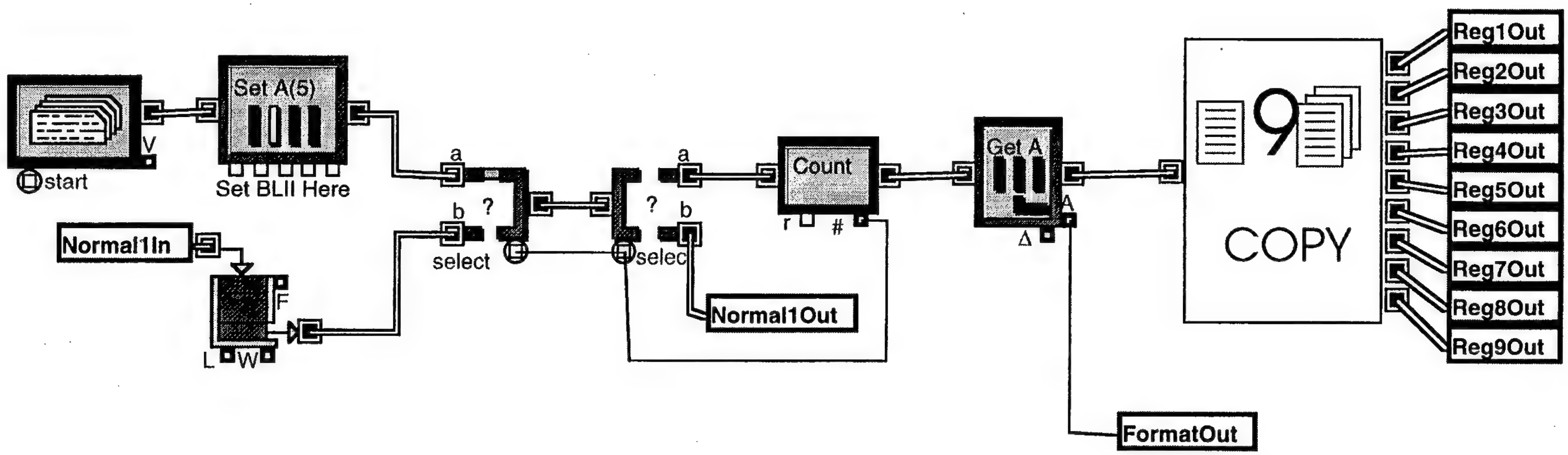
Con1In
F
L
W
a
b
c
demand
regenerate 3
C
TRIP
Con1Out
Con2Out
Con3Out
Con4Out
Con5Out
Con6Out
Con7Out
Con8Out
Con9Out

Structure of Begin Initialize (ATMNETWORK.LIX)

Icon of block Begin Initialize



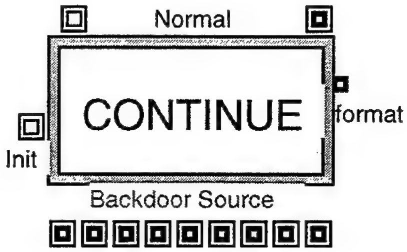
Structure of Begin Initialize (ATMNETWORK.LIX)



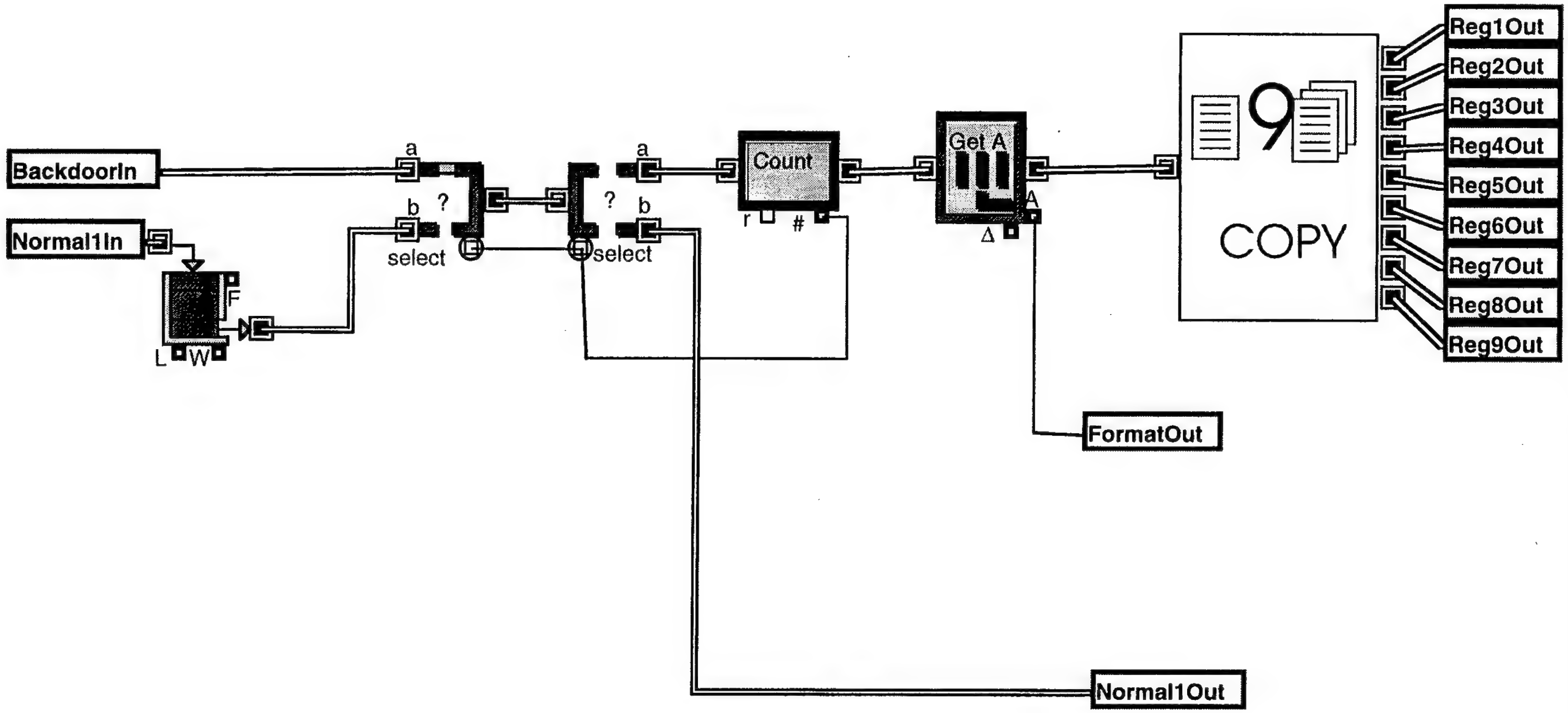
Formats:	Attributes:	Valid Values:
0 : FDDI	Workgroup	3,4,7
1 : ATM OC-12	Building	2,4,6,7
2 : ATM OC-3	Area Distribution Node	0,2,4,5,6
3 : ATM DS-3	Backbone	0,1,2,4,5,6
4 : Classic IP	External	2,4,6
5 : Gb E/N		
6 : Fast E/N		
7 : E/N		

Structure of Continue Initialize (ATMNETWORK.LIX)

Icon of block Continue Initialize



Structure of Continue Initialize (ATMNETWORK.LIX)

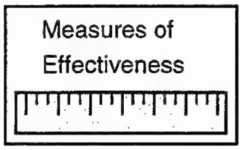


IMPORTANT!

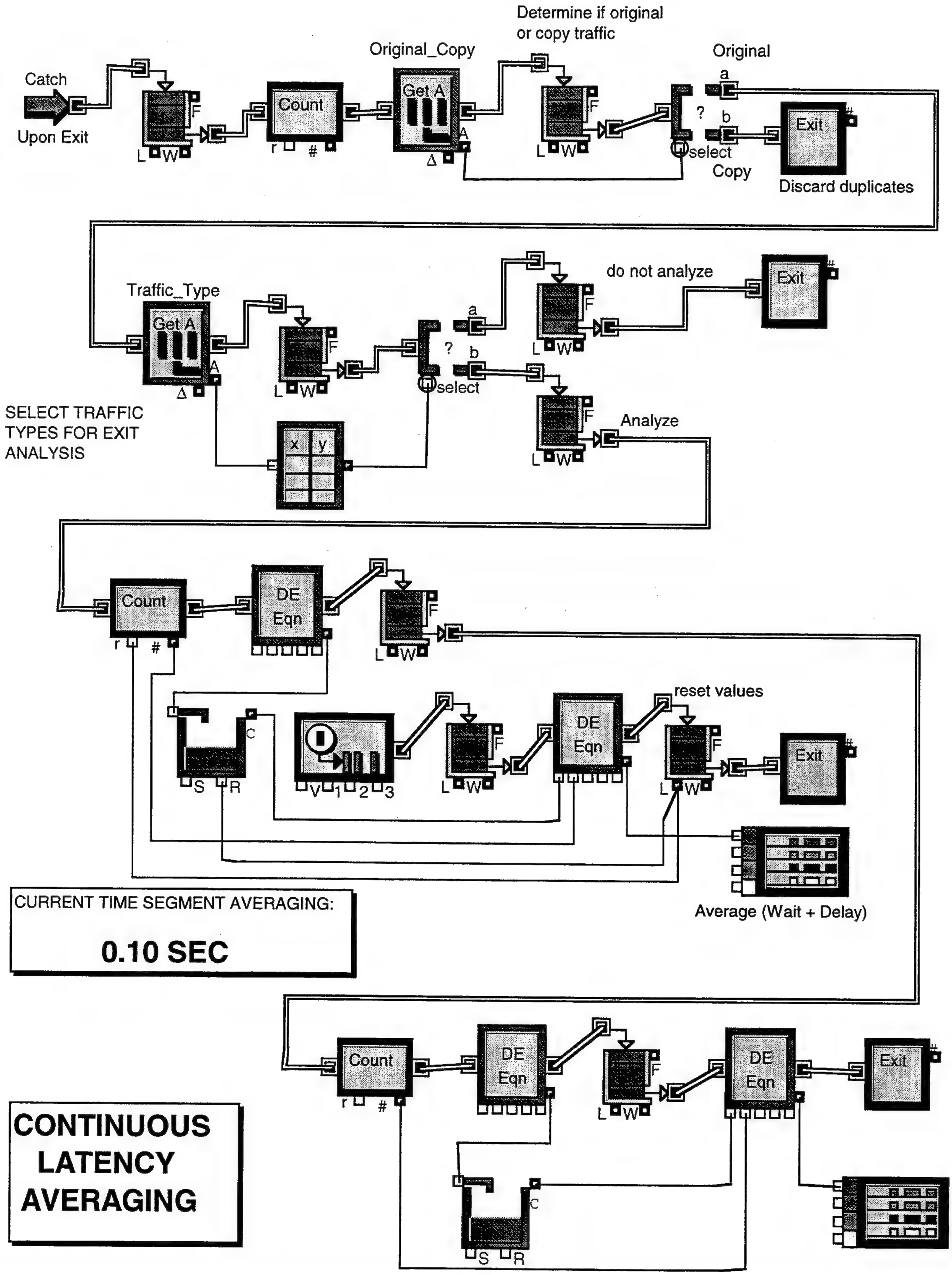
When inserting this block into a model, you must remember to select the appropriate attribute to use in the GET ATTRIBUTE block above. Must choose the correct level that the block resides in, such as Backbone or Workgroup

Structure of Measures Of Effectiveness (ATMNETWORK.LIX)

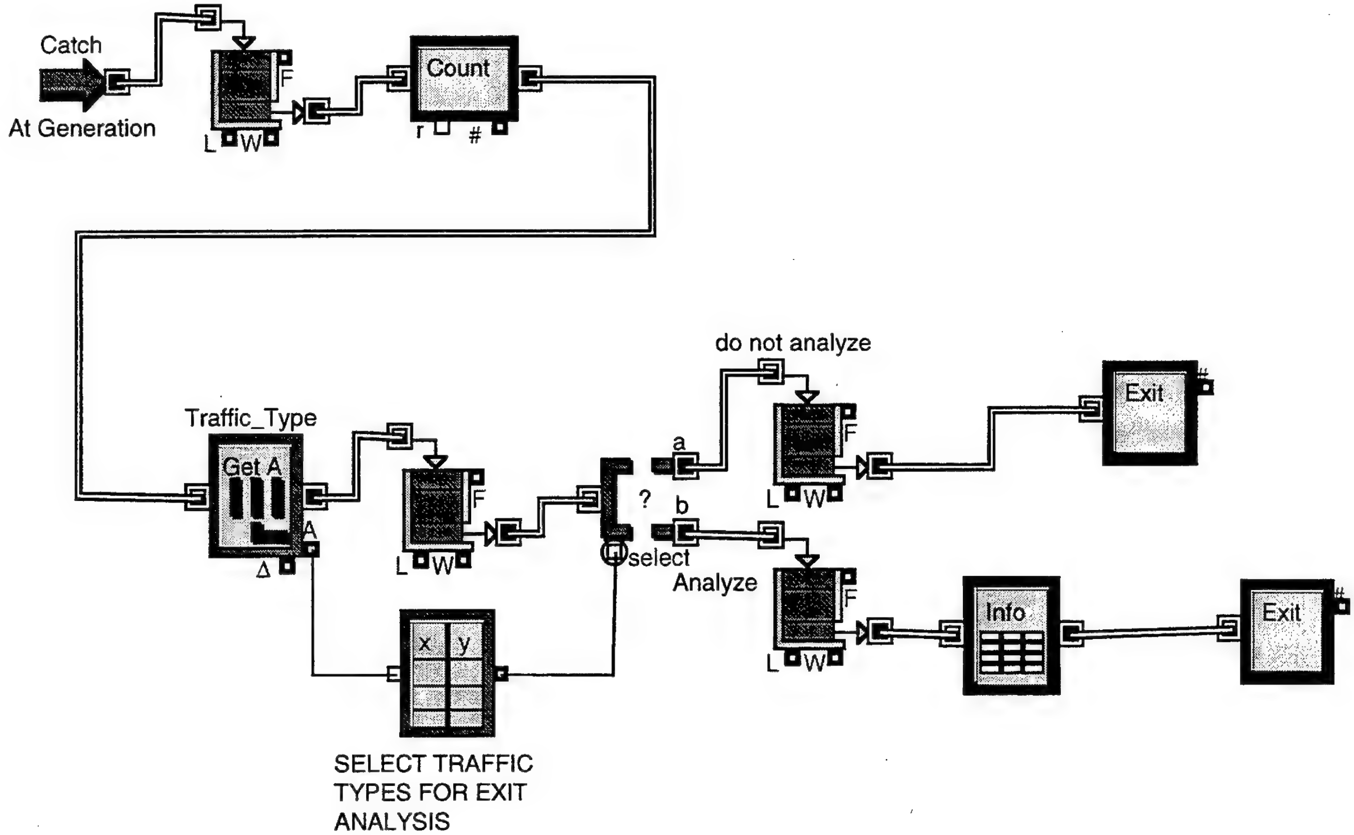
Icon of block Measures Of Effectiveness



Structure of Measures Of Effectiveness (ATMNETWORK.LIX)



Structure of Measures Of Effectiveness (ATMNETWORK.LIX)



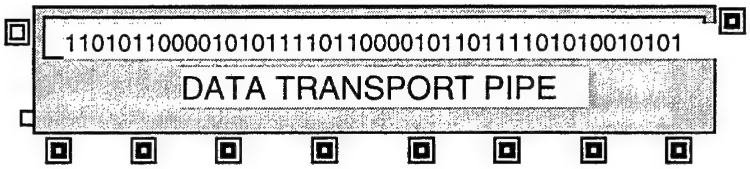
APPENDIX E. EXTEND FUNCTIONAL BLOCKS

The Extend blocks in this appendix are the functional hierarchical blocks used at various locations throughout the infrastructure of the BLII. These blocks each perform a specific function relating to the processing of network traffic. The following hierarchical blocks are contained in this appendix:

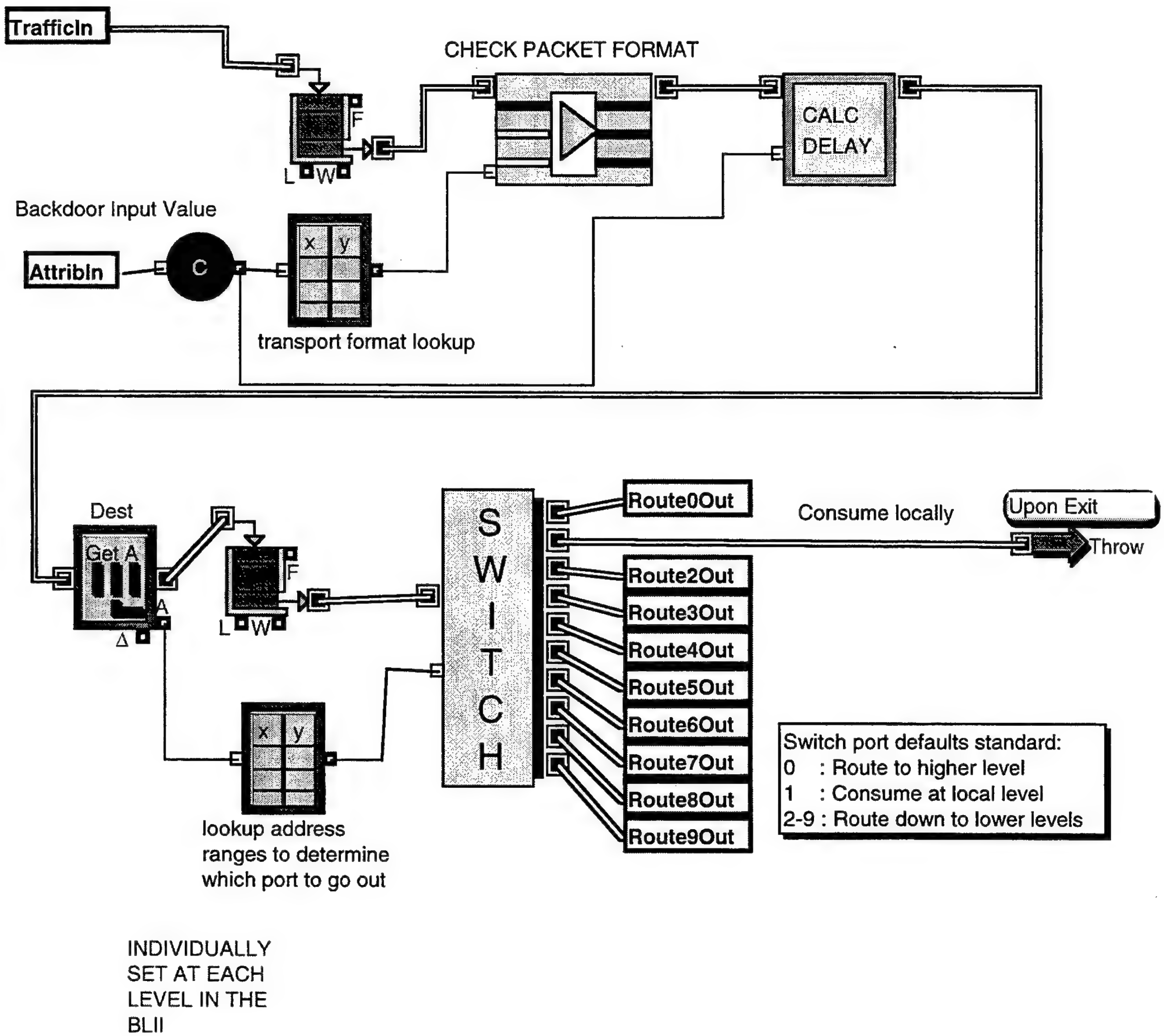
- Data Pipe
- Check Packet Format
- Calculate Delay
- ADN Traffic Generator
- Building Traffic Generator
- Workgroup Traffic Generator
- Determine Destination
- Command VTC
- Backbone Network Services

Structure of Data Pipe (ATMNETWORK.LIX)

Icon of block Data Pipe



Structure of Data Pipe (ATMNETWORK.LIX)

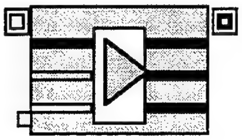


CUSTOMIZED FOR MODEL ADDRESS:

BACKBONE

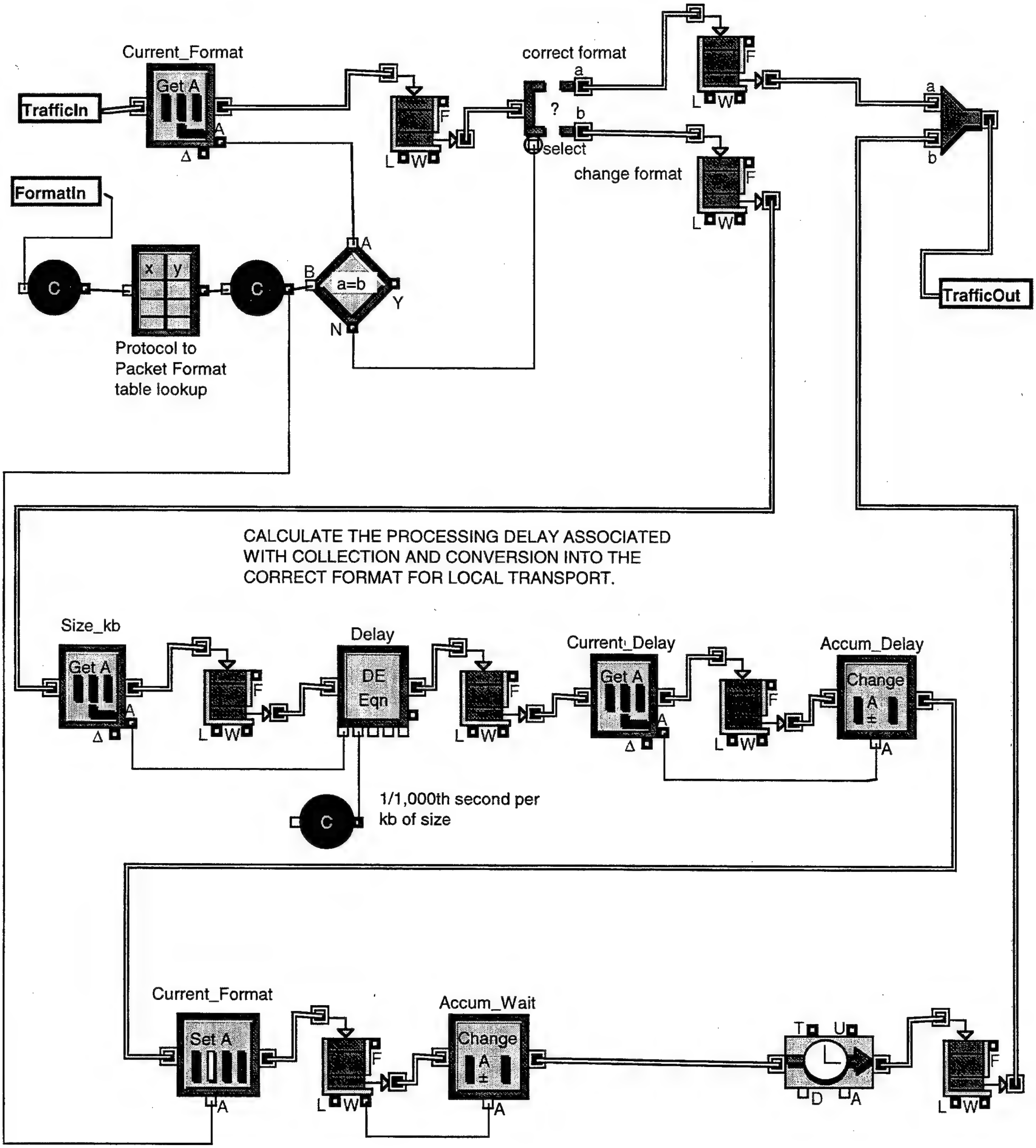
Structure of Check Packet Format (COMPARE FORMAT PARTS.LIX)

Icon of block Check Packet Format



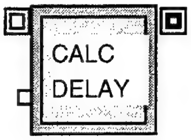
Structure of Check Packet Format (COMPARE FORMAT PARTS.LIX)

COMPARE THE CURRENT FORMAT OF THE INCOMING TRAFFIC TO DETERMINE IF IT IS IN THE FORMAT REQUIRED FOR THIS LEVEL TRANSPORT PIPE. IF NOT, CONVERT TO THE RIGHT FORMAT.

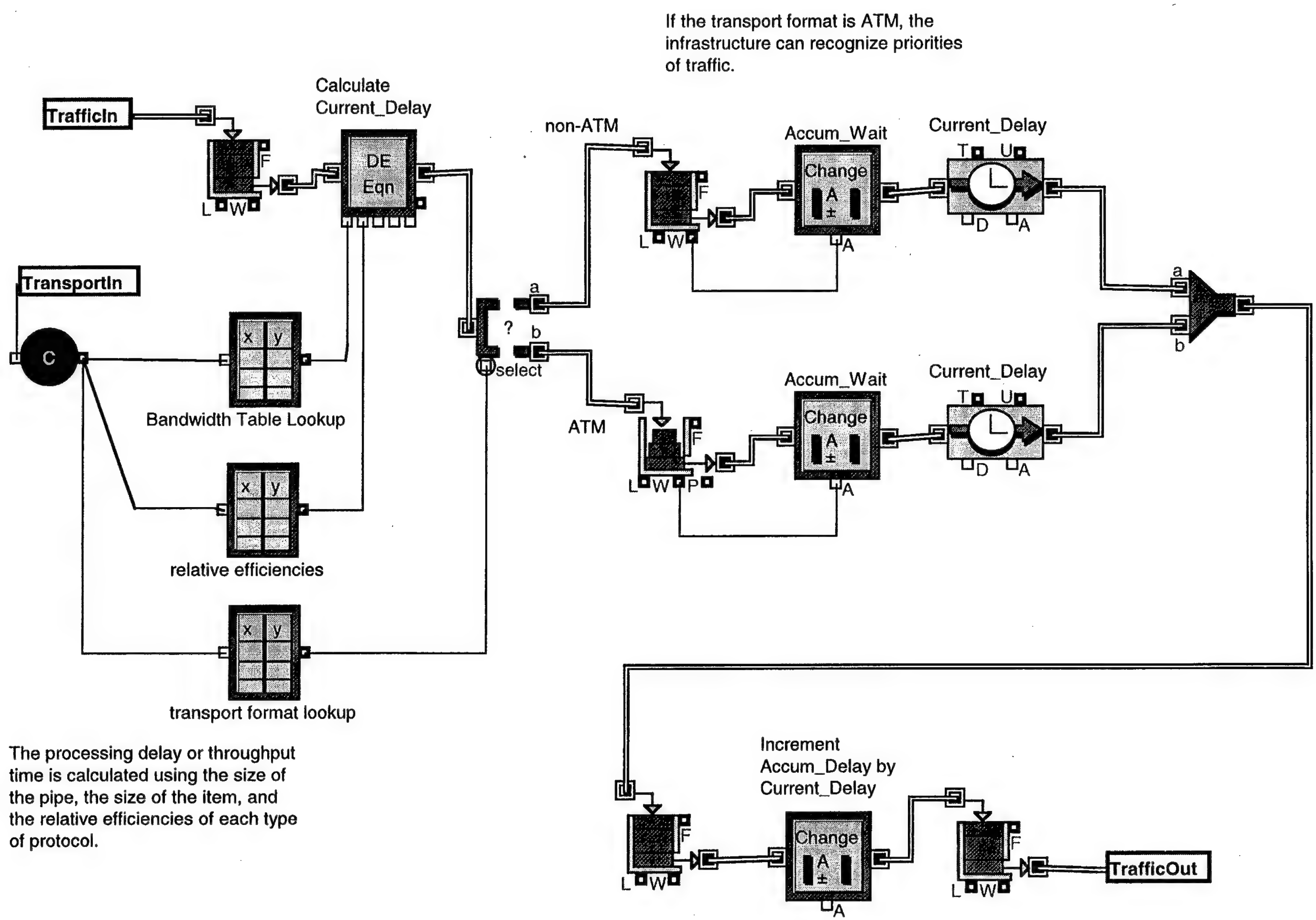


Structure of Process Delay (ATMNETWORK.LIX)

Icon of block Process Delay



Structure of Process Delay (ATMNETWORK.LIX)



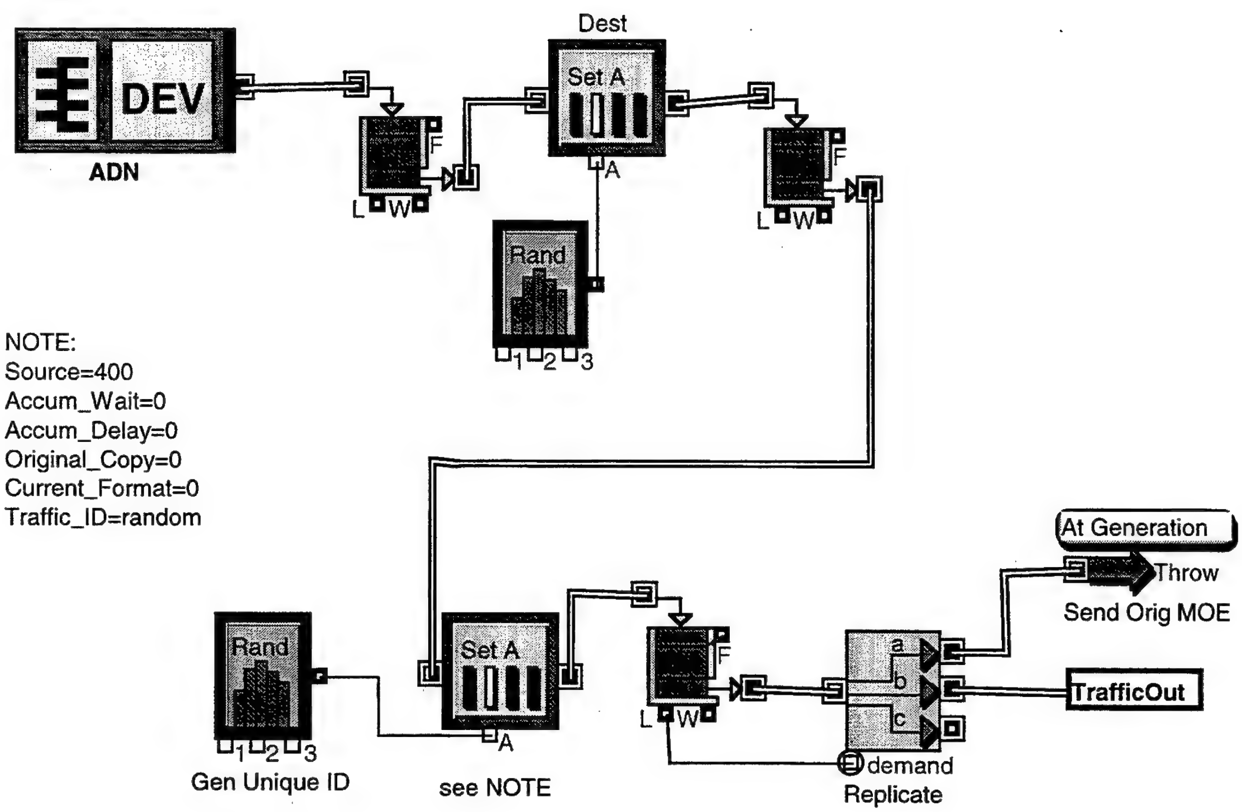
Structure of GENERATOR, ADN (ATMNETWORK.LIX)

Icon of block GENERATOR, ADN

Structure of GENERATOR, ADN (ATMNETWORK.LIX)

CUSTOMIZED FOR ADDRESS:

400

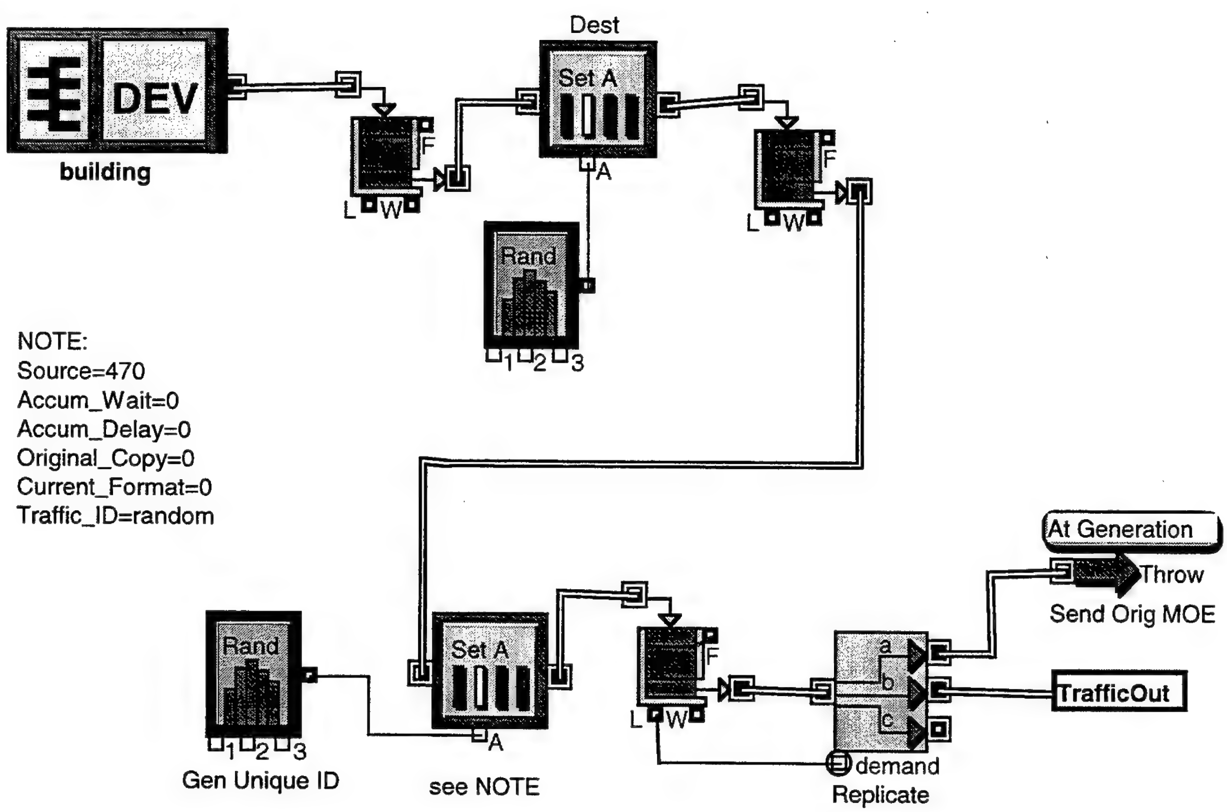


Icon of block GENERATOR, building

Structure of GENERATOR, building (ATMNETWORK.LIX)

CUSTOMIZED FOR ADDRESS:

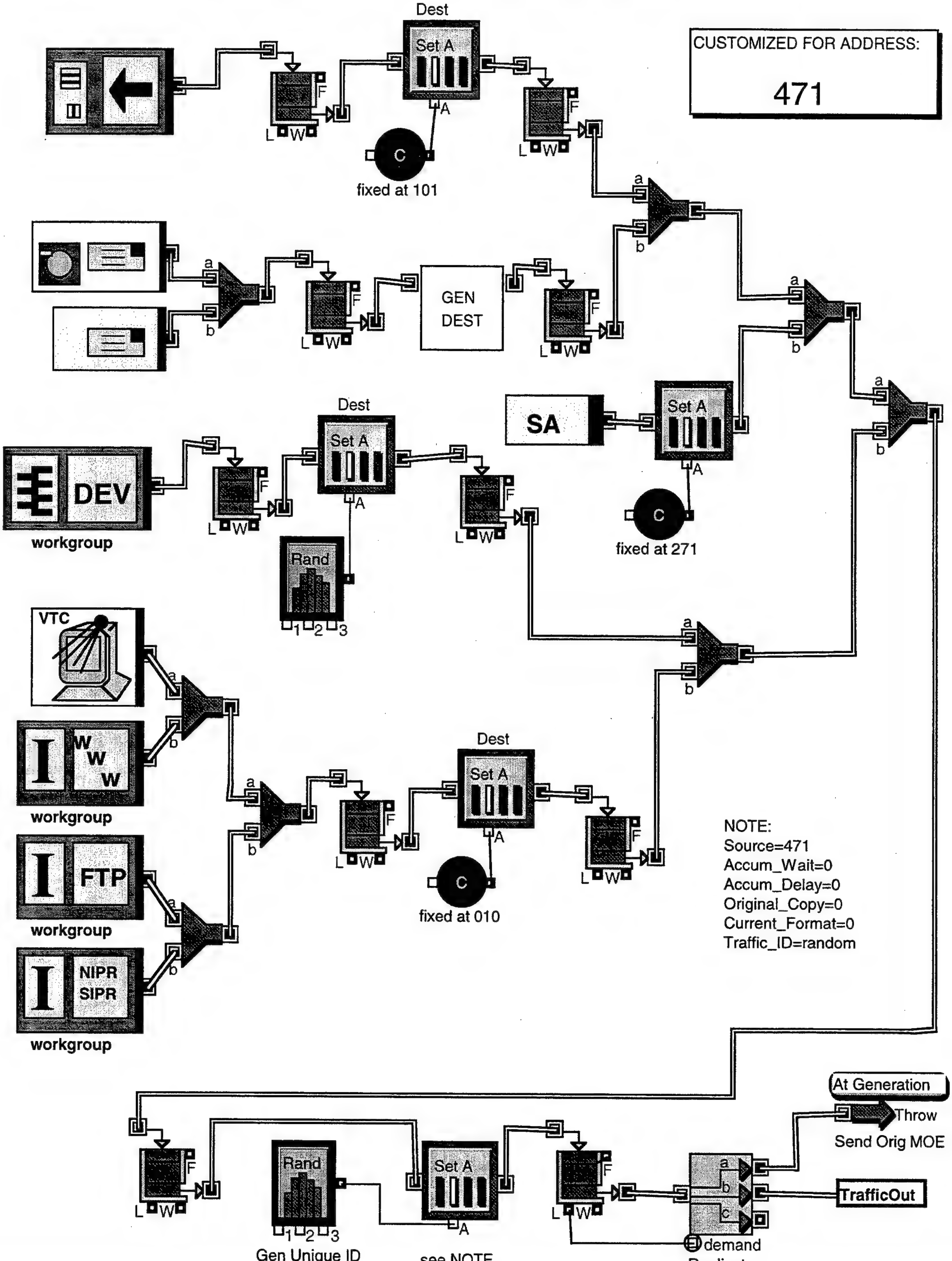
470



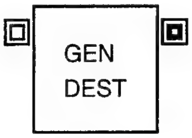
NOTE:
Source=470
Accum_Wait=0
Accum_Delay=0
Original_Copy=0
Current_Format=0
Traffic_ID=random

Icon of block GENERATOR, workgroup

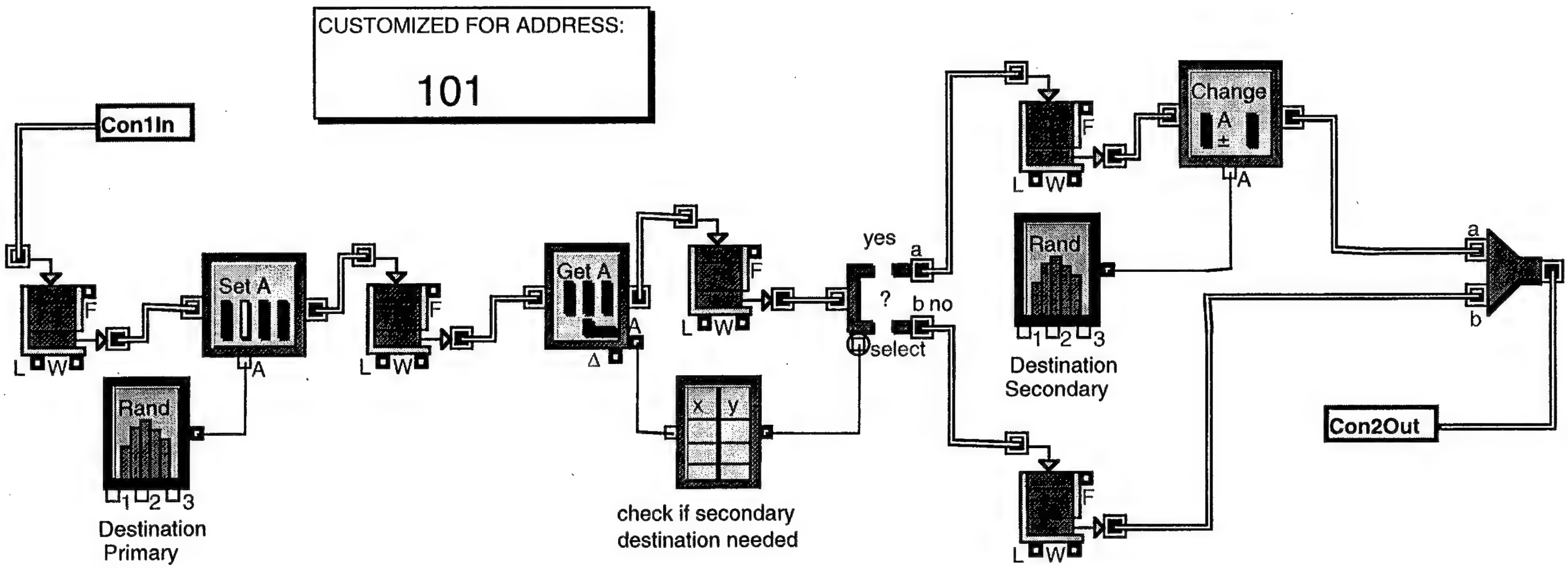
Structure of GENERATOR, workgroup (ATMNETWORK.LIX)



Icon of block Determine Destination

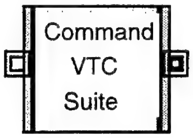


Structure of Determine Destination

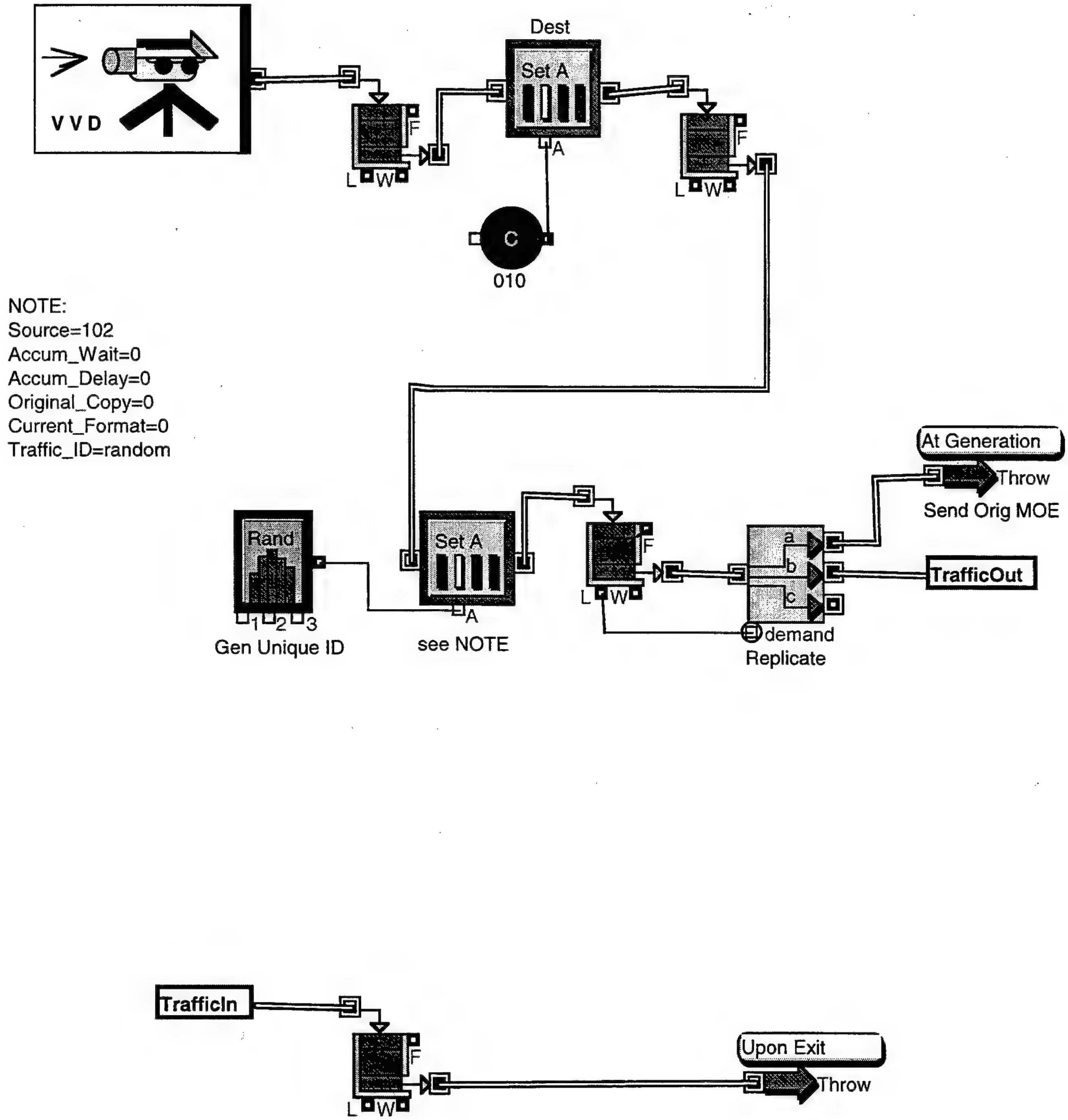


MAKE CHANGES IN THE DISTRIBUTION BLOCK BASED ON THE SOURCE BLOCK AND THE TYPE OF TRAFFIC. MUST BE CUSTOMIZED FOR EACH INDIVIDUAL ADDRESS.

Icon of block Command VTC Suite



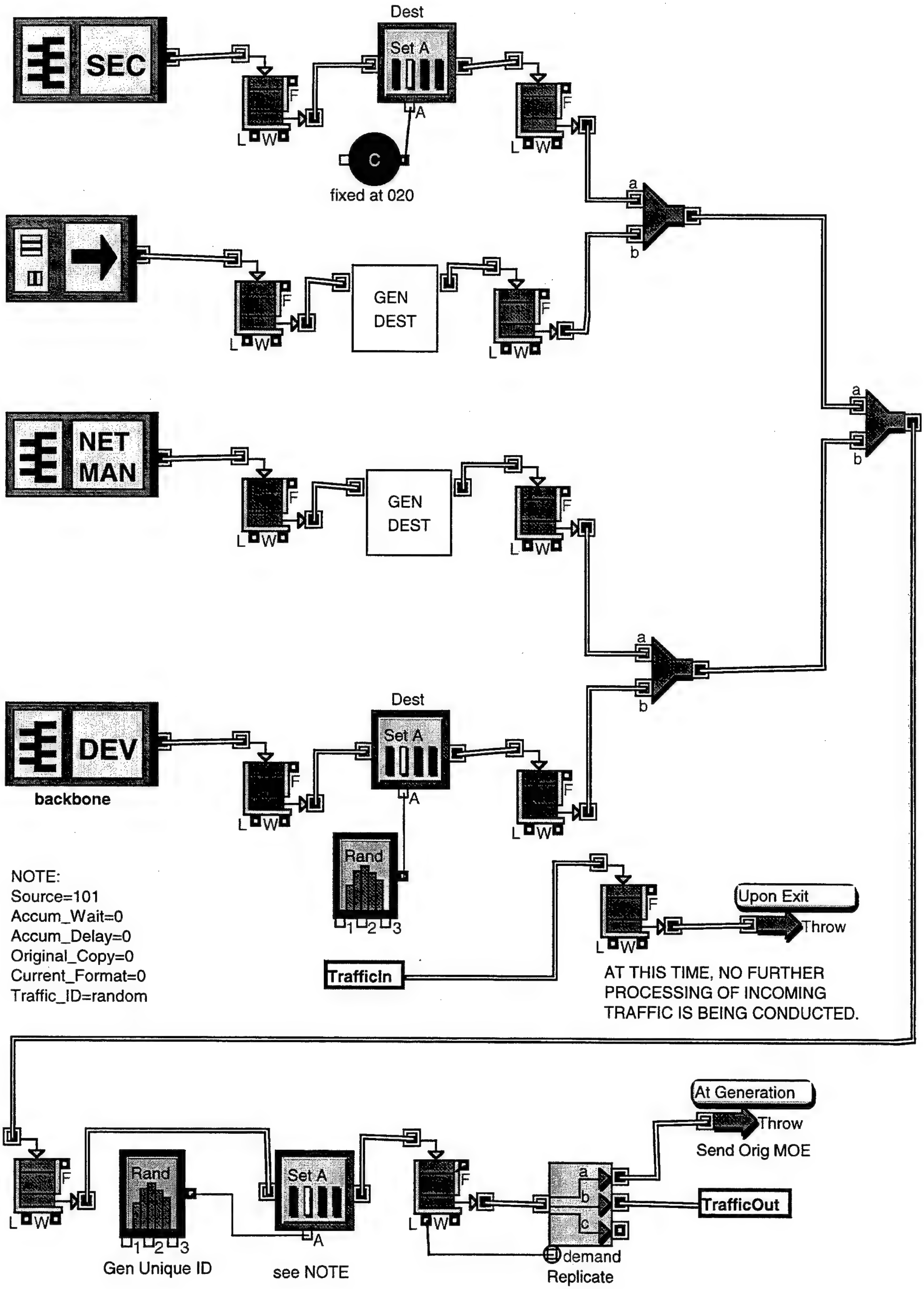
FORMAL STUDIO SETUP FOR VIRTUAL STAFFING, FULL-MOTION VTC, AND DISTANCE LEARNING.



AT THIS TIME, NO FURTHER PROCESSING OF INCOMING VTC TRAFFIC IS BEING CONDUCTED.

Icon of block Network Services

Structure of Network Services



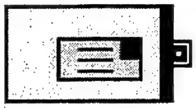
APPENDIX F. EXTEND TRAFFIC GENERATOR BLOCKS

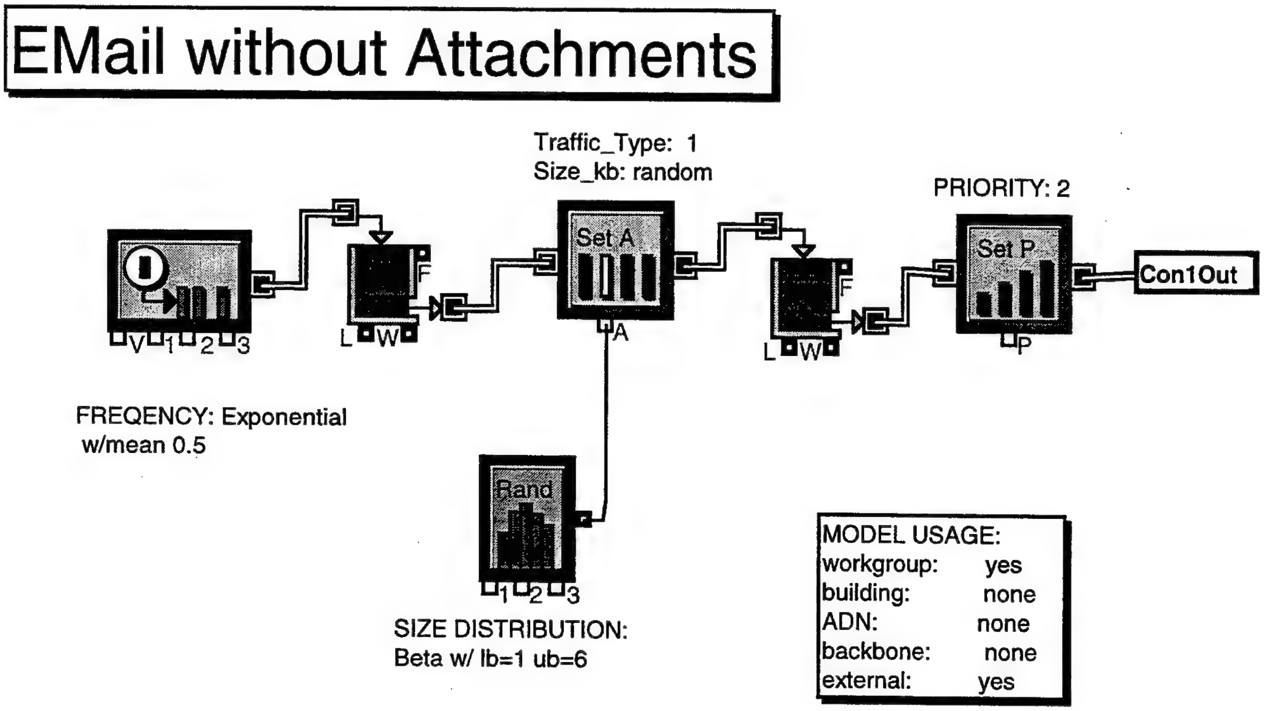
The Extend blocks in this appendix are the functional hierarchical blocks used at various locations throughout the infrastructure of the BLII. These blocks network traffic at a frequency and size identified by the respective distributions and parameters included in each block. The following hierarchical blocks are contained in this appendix:

- Email with Attachments
- Command VTC
- Email without Attachments
- Desktop VTC
- Inter-device Communications (workgroup)
- Inter-device Communications (building)
- Inter-device Communications (ADN)
- Network Management Applications
- Internet NIPR/SIPR (workgroup)
- Internet NIPR/SIPR (external)
- Internet FTP (setup)
- Internet FTP (download)
- Internet Commercial Surfing (workgroup)
- Internet Commercial Surfing (external)
- Network Resource Request
- Network Resource Response
- Network Security
- Sensitivity Analysis

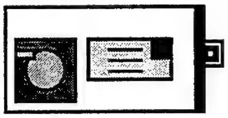
Structure of EMail without (NETWORK TRAFFIC GENERATORS.LIX)

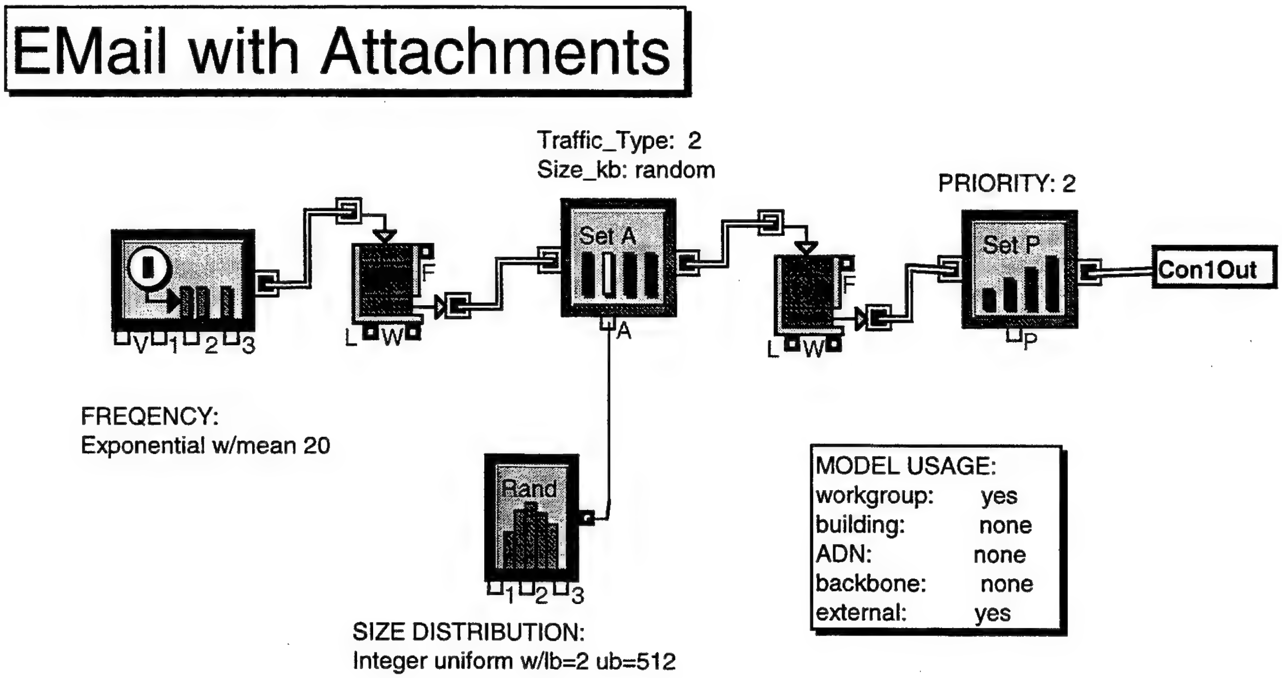
Icon of block EMail without




EMail without Attachments
Traffic_Type: 1
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
Rand
FREQENCY: Exponential
w/mean 0.5
SIZE DISTRIBUTION:
Beta w/ lb=1 ub=6
MODEL USAGE:
workgroup: yes
building: none
ADN: none
backbone: none
external: yes

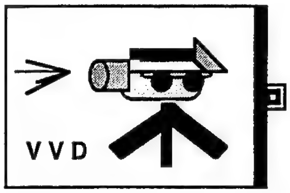
Structure of EMail with Attachments (NETWORK TRAFFIC GENERATORS.LIX)
Icon of block EMail with Attachments



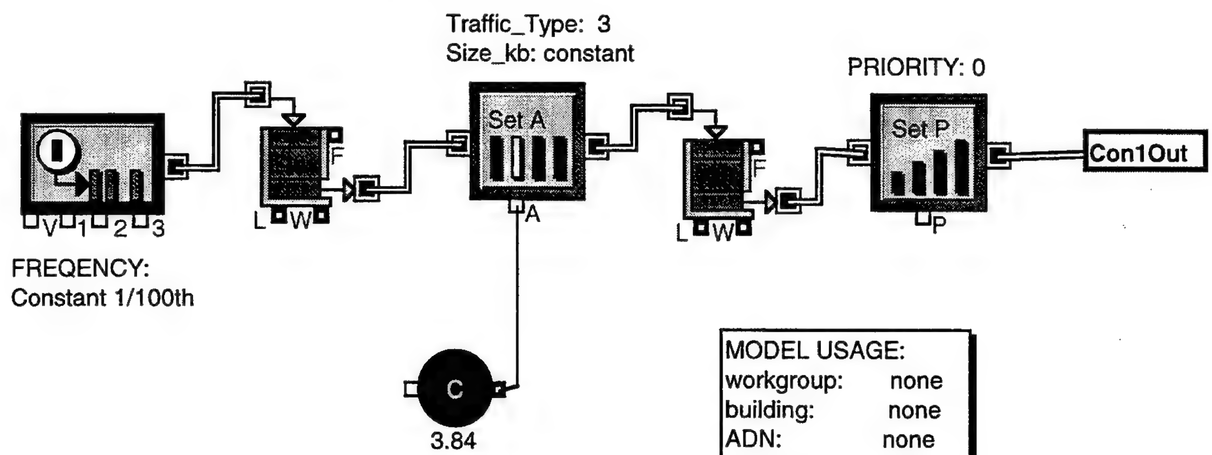

EMail with Attachments
Traffic_Type: 2
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
F
L W
A
P
FREQENCY:
Exponential w/mean 20
Rand
1 2 3
SIZE DISTRIBUTION:
Integer uniform w/lb=2 ub=512
MODEL USAGE:
workgroup: yes
building: none
ADN: none
backbone: none
external: yes

Structure of Command VTC (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Command VTC



Command-Level VTC Suite

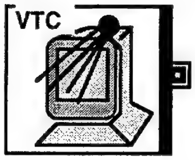


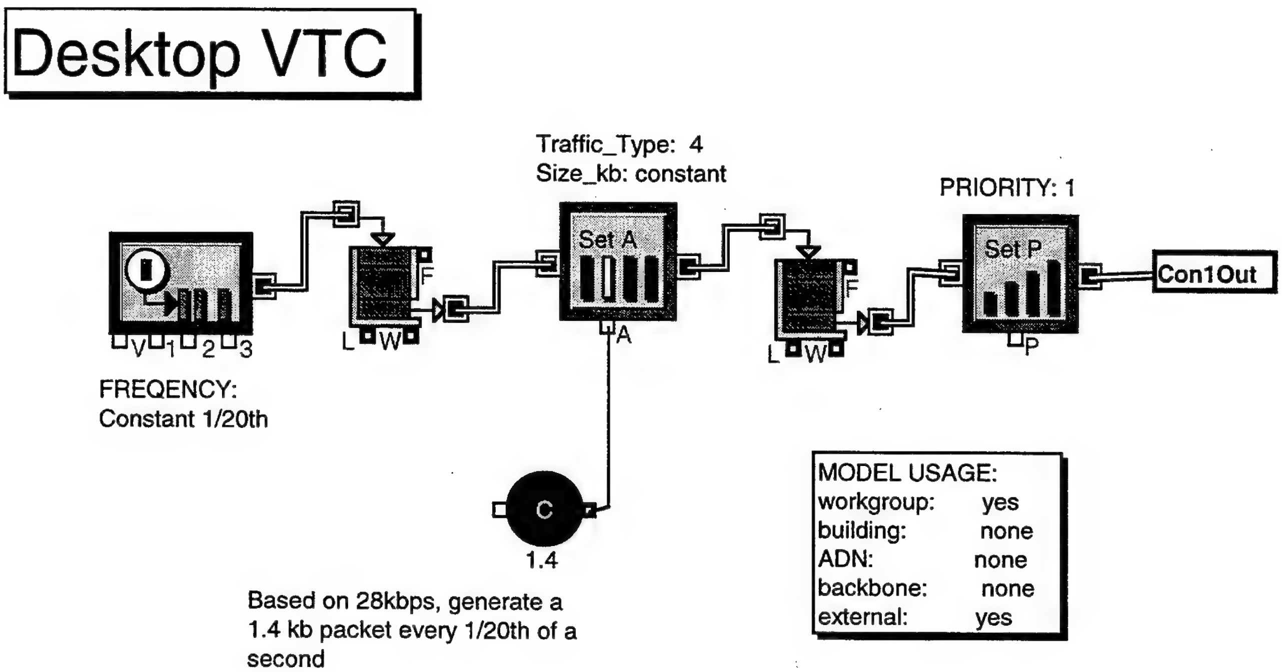
MODEL USAGE:	
workgroup:	none
building:	none
ADN:	none
backbone:	yes
external:	yes

Based on required bandwidth requirement of 384 kbps, send packets as 3.84 kb every 1/100th second

Structure of Desktop VTC (NETWORK TRAFFIC GENERATORS.LIX)

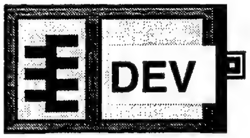
Icon of block Desktop VTC



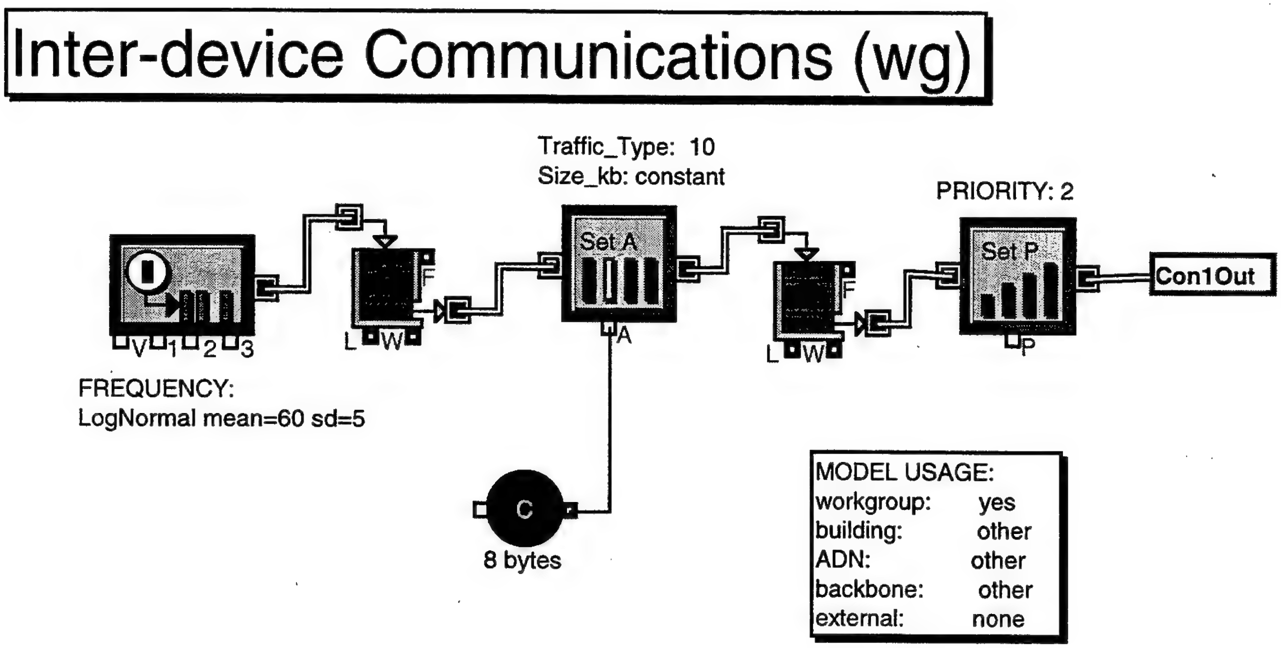

Desktop VTC
Traffic_Type: 4
Size_kb: constant
PRIORITY: 1
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
FREQENCY:
Constant 1/20th
C
1.4
Based on 28kbps, generate a 1.4 kb packet every 1/20th of a second
MODEL USAGE:
workgroup: yes
building: none
ADN: none
backbone: none
external: yes

Structure of Inter-device Communication (wg) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Inter-device Communication (wg)

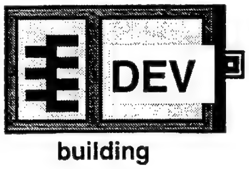


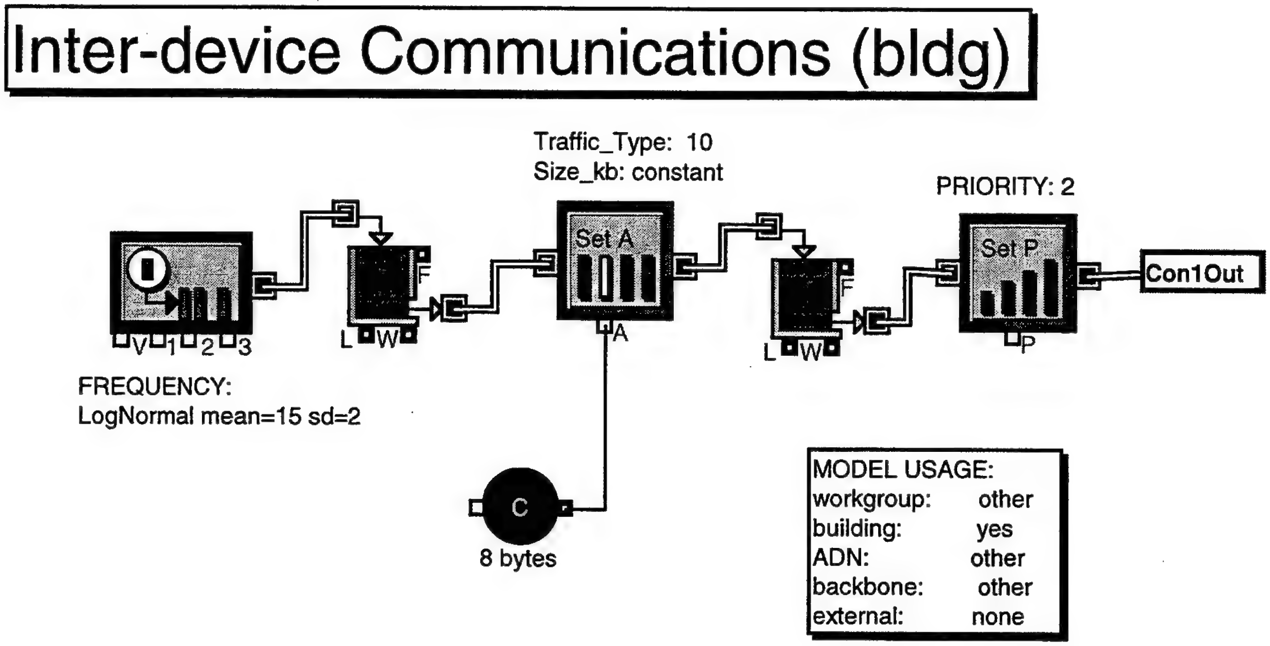
workgroup


Inter-device Communications (wg)
Traffic_Type: 10
Size_kb: constant
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
FREQUENCY:
LogNormal mean=60 sd=5
C
8 bytes
MODEL USAGE:
workgroup: yes
building: other
ADN: other
backbone: other
external: none

Structure of Inter-device Communication (b) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Inter-device Communication (b)





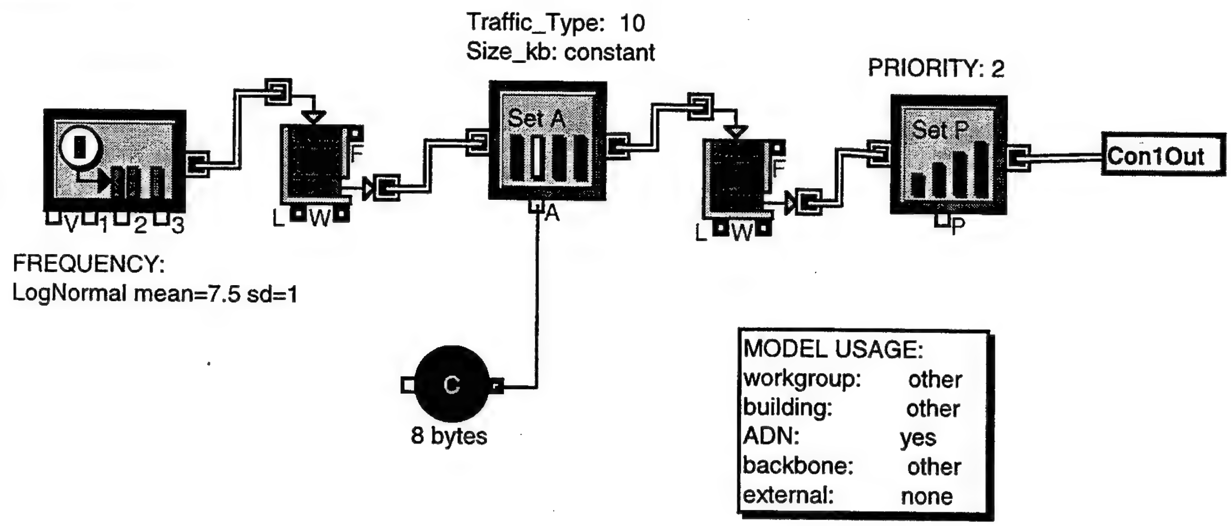
Inter-device Communications (bldg)
Traffic_Type: 10
Size_kb: constant
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
FREQUENCY:
LogNormal mean=15 sd=2
C
8 bytes
MODEL USAGE:
workgroup: other
building: yes
ADN: other
backbone: other
external: none

Structure of Inter-device Communication (dn) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Inter-device Communication (dn)

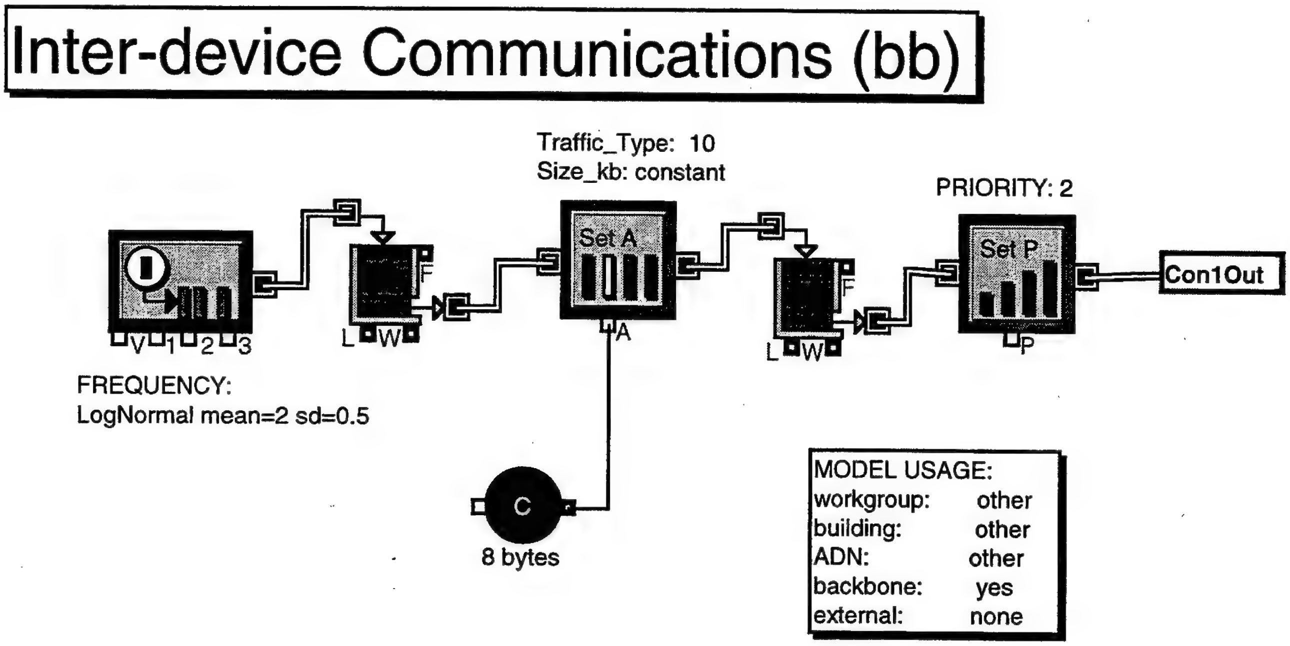
ADN

Inter-device Communications (ADN)



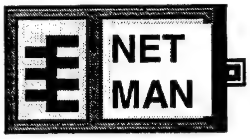
Structure of Inter-device Communication (bb) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Inter-device Communication (bb)

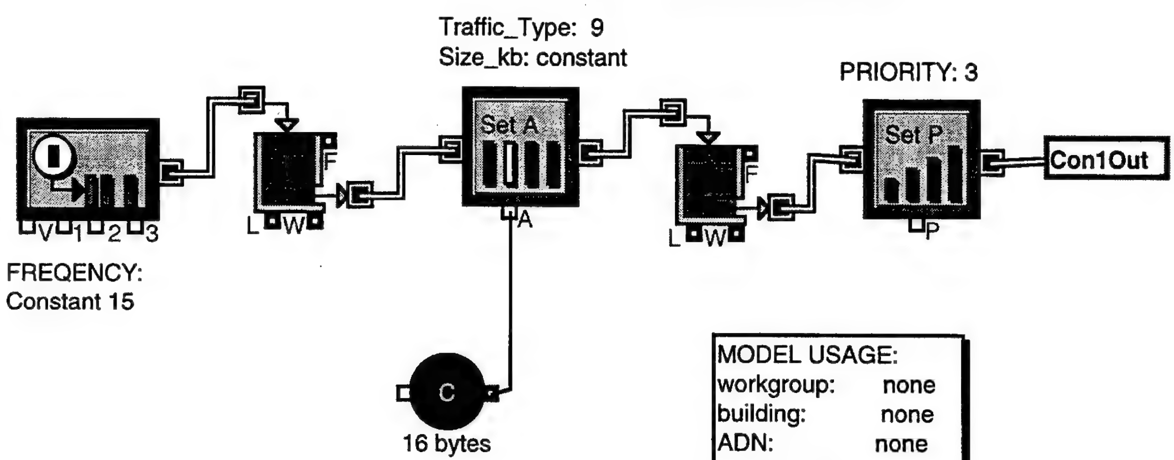

Inter-device Communications (bb)
Traffic_Type: 10
Size_kb: constant
PRIORITY: 2
Set A
Set P
Con1Out
V
1
2
3
L
W
F
A
P
FREQUENCY:
LogNormal mean=2 sd=0.5
C
8 bytes
MODEL USAGE:
workgroup: other
building: other
ADN: other
backbone: yes
external: none

Structure of Network Management Apps (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Network Management Apps



Network Management Applications



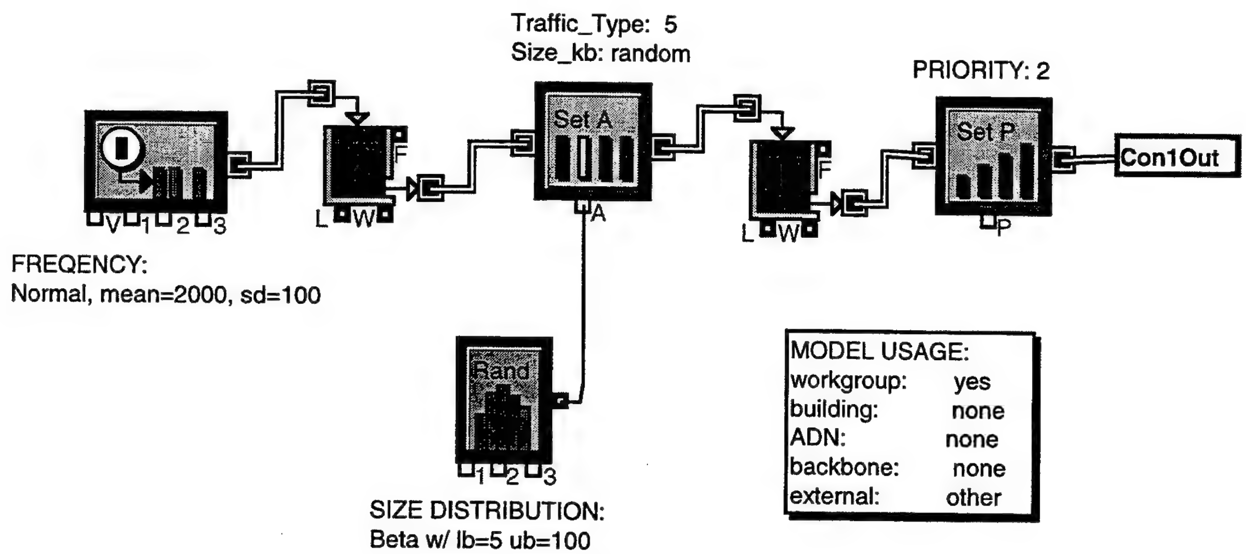
MODEL USAGE:	
workgroup:	none
building:	none
ADN:	none
backbone:	yes
external:	none

NOTE: NetMan polling is only generated at the backbone level. We received by the destination, it is "turned around" and send back as a response, therefore no generators exist at other levels of the BLII.

Structure of Internet NIPR/SIPR (wg) (NETWORK TRAFFIC GENERATORS.LIX)

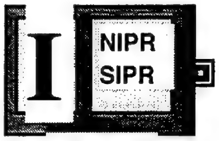
Icon of block Internet NIPR/SIPR (wg)

Internet, NIPRNET/SIPRNET (workgroup)



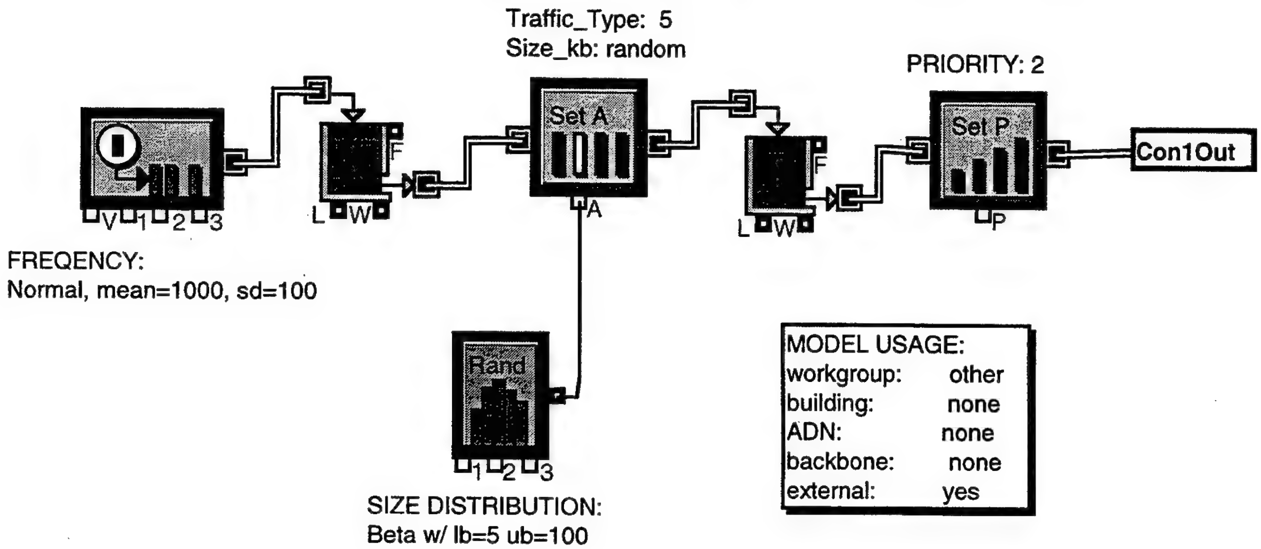
Structure of Internet NIPR/SIPR (ext) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Internet NIPR/SIPR (ext)



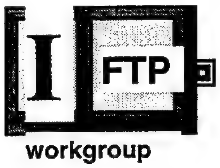
external

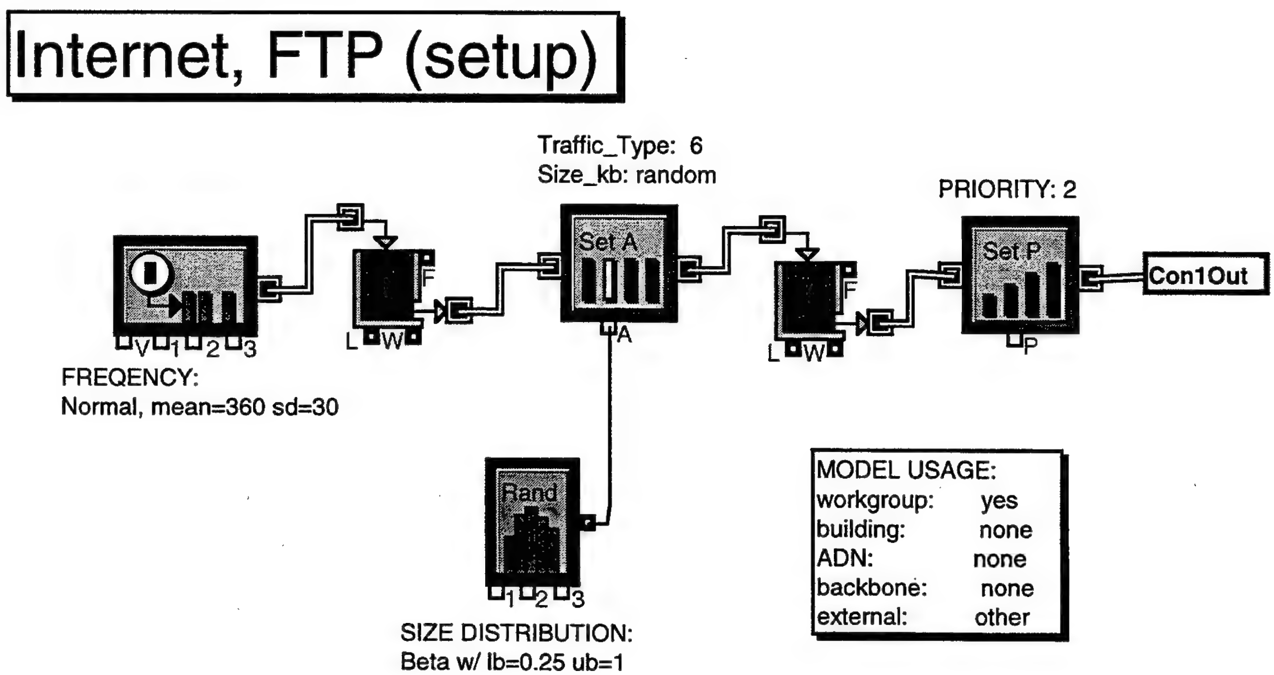
Internet, NIPRNET/SIPRNET (external)



Structure of Internet FTP (setup) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Internet FTP (setup)

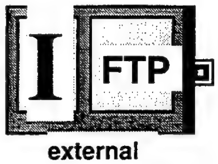


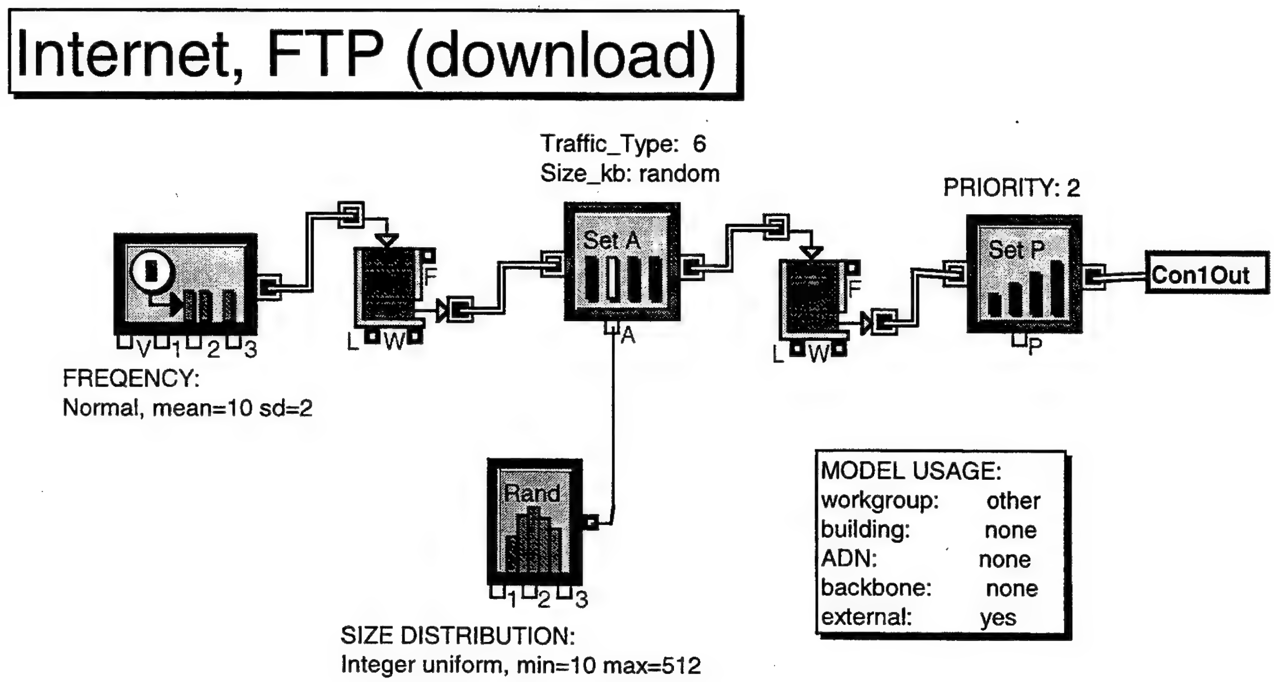


Internet, FTP (setup)
Traffic_Type: 6
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
FREQENCY:
Normal, mean=360 sd=30
Rand
1 2 3
SIZE DISTRIBUTION:
Beta w/ lb=0.25 ub=1
MODEL USAGE:
workgroup: yes
building: none
ADN: none
backbone: none
external: other

Structure of Internet FTP (download) (NETWORK TRAFFIC GENERATORS.LIX)

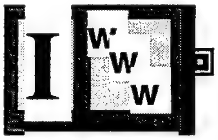
Icon of block Internet FTP (download)




Internet, FTP (download)
Traffic_Type: 6
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
FREQENCY:
Normal, mean=10 sd=2
Rand
1 2 3
SIZE DISTRIBUTION:
Integer uniform, min=10 max=512
MODEL USAGE:
workgroup: other
building: none
ADN: none
backbone: none
external: yes

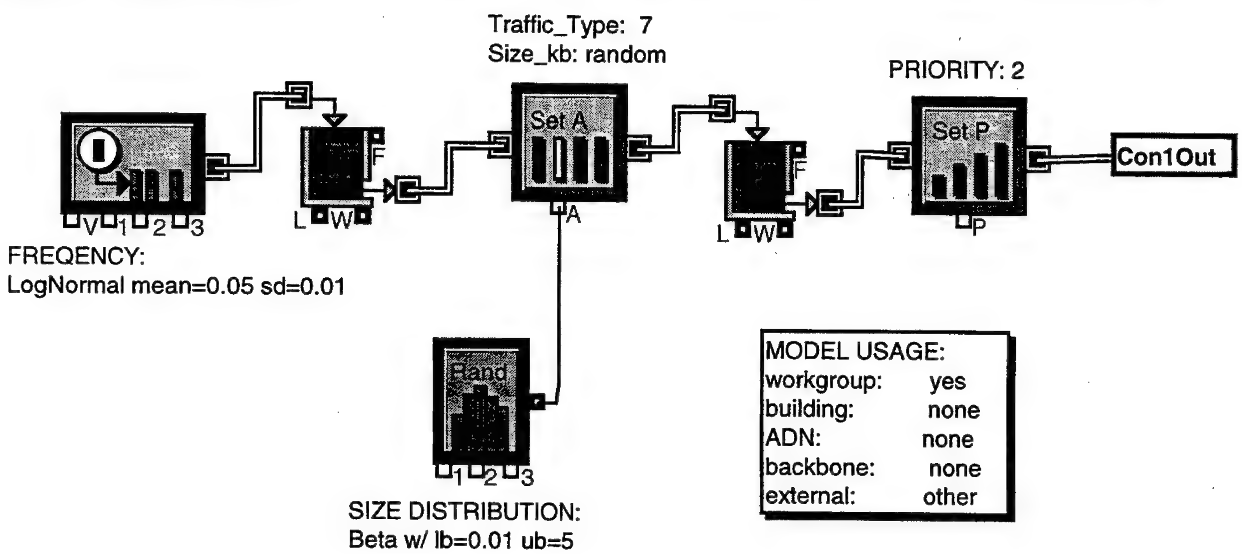
Structure of Internet Com'l Surfing (wg) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Internet Com'l Surfing (wg)



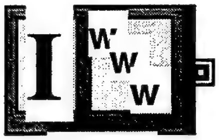
workgroup

Internet, Commercial Surfing (workgroup)



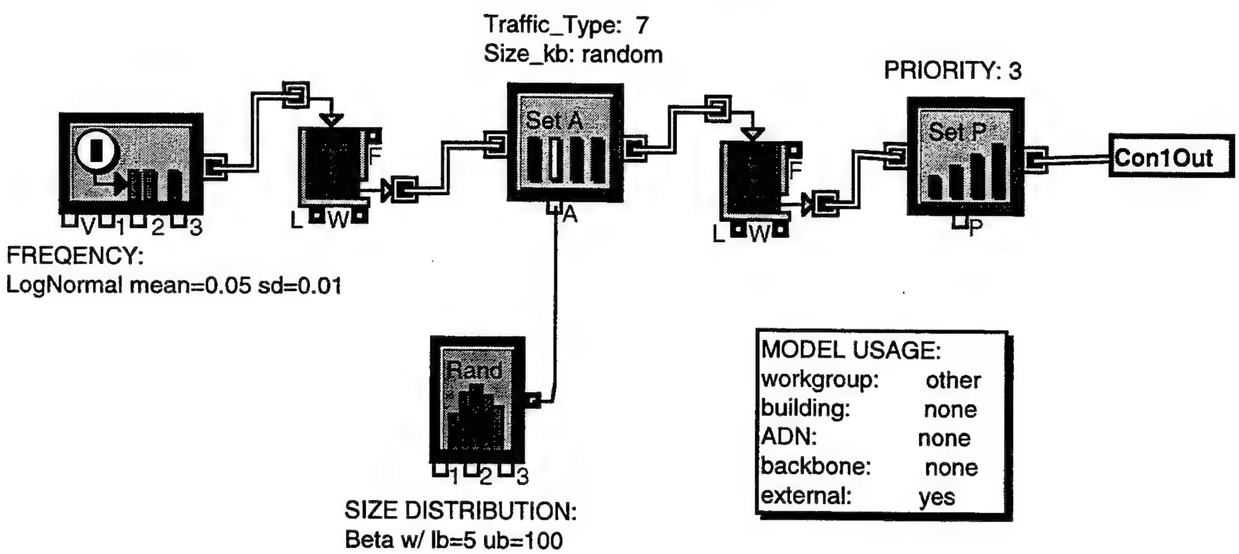
Structure of Internet Com'l Surfing (ext) (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Internet Com'l Surfing (ext)



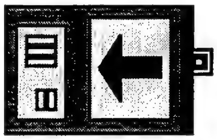
external

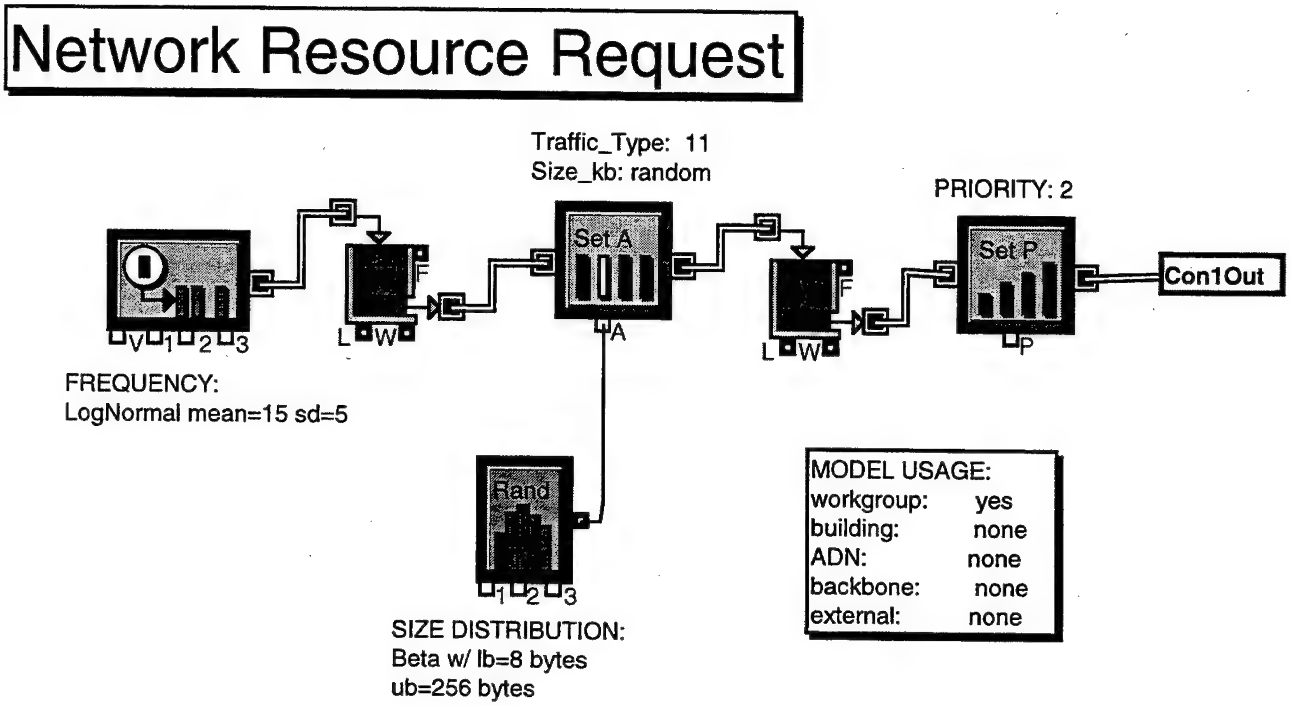
Internet, Commercial Surfing (external)



Structure of Network Resource Request (NETWORK TRAFFIC GENERATORS.LIX)

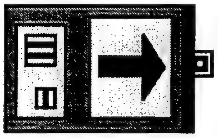
Icon of block Network Resource Request

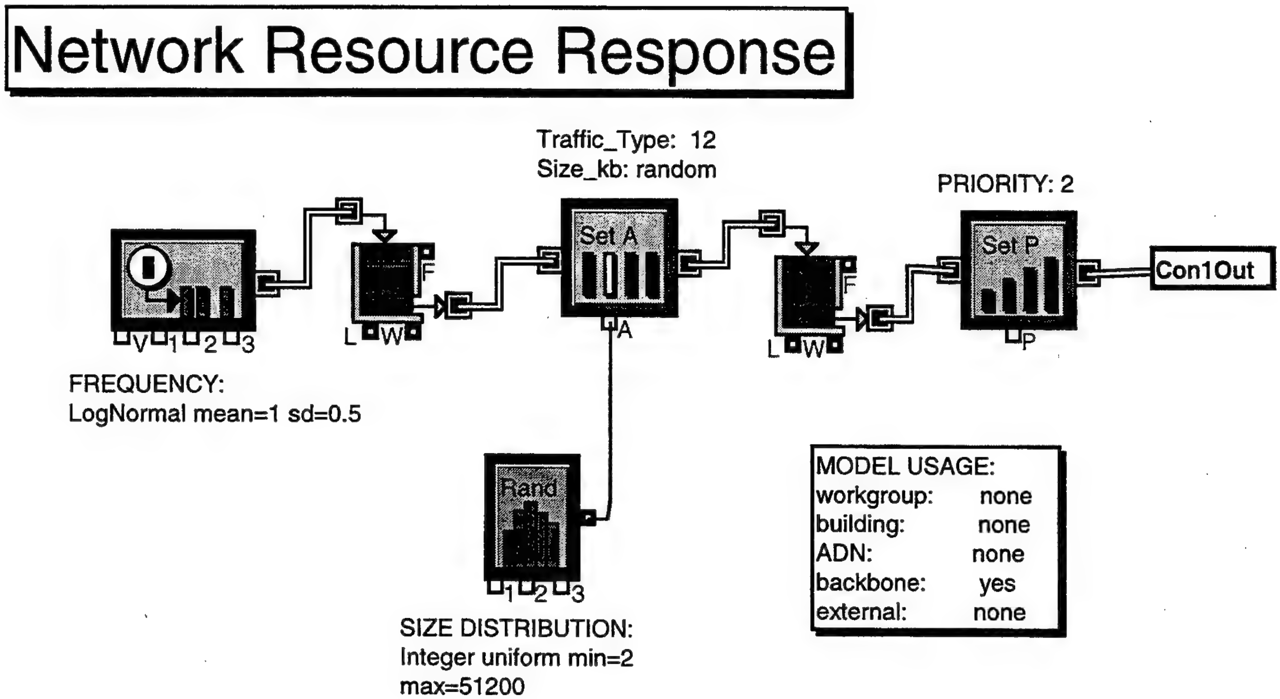



Network Resource Request
Traffic_Type: 11
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
V 1 2 3
L W F
A
P
Rand
1 2 3
FREQUENCY:
LogNormal mean=15 sd=5
SIZE DISTRIBUTION:
Beta w/ lb=8 bytes
ub=256 bytes
MODEL USAGE:
workgroup: yes
building: none
ADN: none
backbone: none
external: none

Structure of Network Resource Response (NETWORK TRAFFIC GENERATORS.LIX)

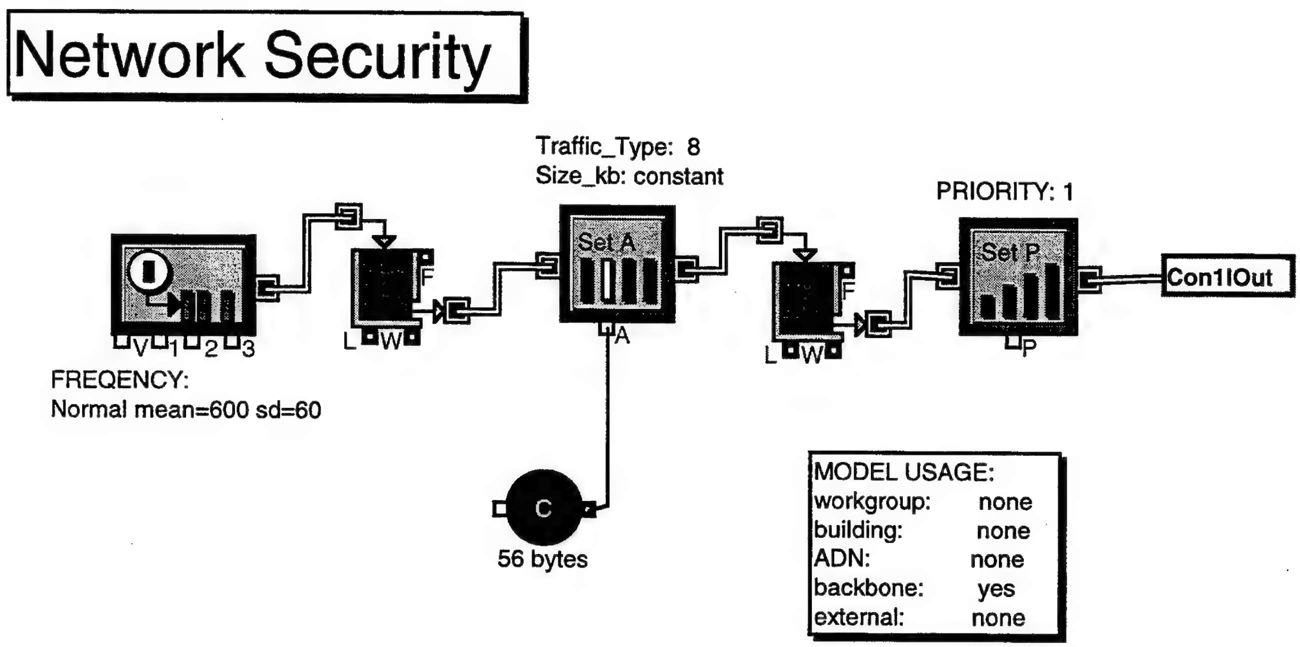
Icon of block Network Resource Response




Network Resource Response
Traffic_Type: 12
Size_kb: random
PRIORITY: 2
Set A
Set P
Con1Out
FREQUENCY:
LogNormal mean=1 sd=0.5
Rand
SIZE DISTRIBUTION:
Integer uniform min=2
max=51200
MODEL USAGE:
workgroup: none
building: none
ADN: none
backbone: yes
external: none

Structure of Network Security (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Network Security

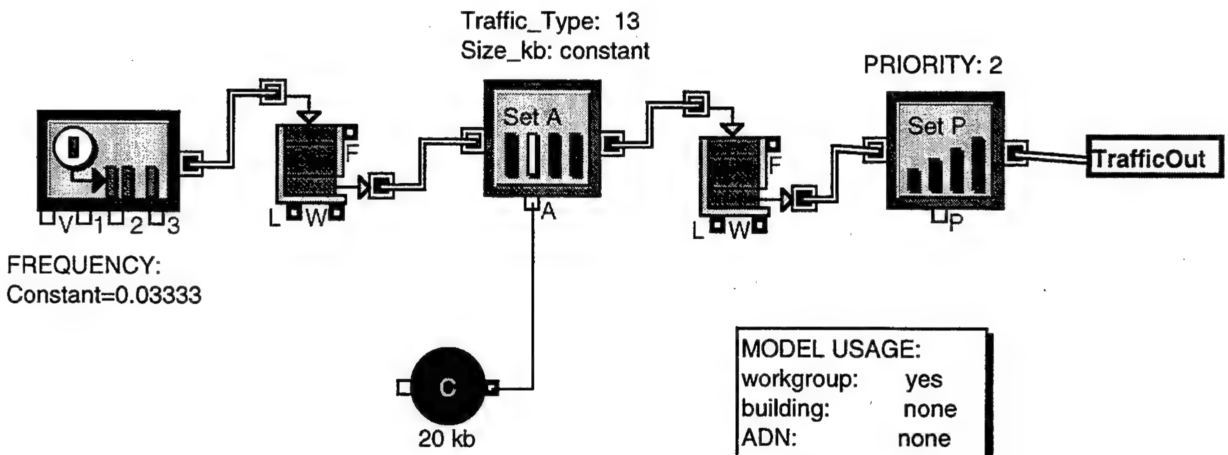


Network Security
Traffic_Type: 8
Size_kb: constant
PRIORITY: 1
Set A
Set P
Con1IOut
V 1 2 3
L W F
A
P
FREQENCY:
Normal mean=600 sd=60
C
56 bytes
MODEL USAGE:
workgroup: none
building: none
ADN: none
backbone: yes
external: none

Structure of Sensitivity Analysis (NETWORK TRAFFIC GENERATORS.LIX)

Icon of block Sensitivity Analysis

Sensitivity Analysis Generator



MODEL USAGE:	
workgroup:	yes
building:	none
ADN:	none
backbone:	none
external:	none

This block will be used to test the sensitivity of the model architectures by varying the overall amount of traffic present. This will simulate an overall traffic increase for the network based on the parameters above, but will not be associated with any particular type of traffic.

PARAMETER SETTINGS FOR SENSITIVITY ANALYSIS:

	NORMAL NETWORK LOAD	MEDIUM WORKLOAD INCREASE	HEAVY WORKLOAD INCREASE
Freq Dist Type	Constant	Constant	Constant
Parameter (sec)	60	0.05	0.0333
Size Dist Type	Constant	Constant	Constant
Parameter (kb)	20	20	20
Effective Congestion Workgroup (Mbps)	Nominal	0.8	1.2
Effective Congestion Building (Mbps)	Nominal	2.4	3.6
Effective Congestion ADN (Mbps)	Nominal	16.8	25.2
Effective Congestion Backbone (Mbps)	Nominal	33.6	50.4

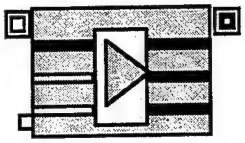
APPENDIX G. EXTEND PACKET LEVEL FUNCTIONAL BLOCKS

The Extend blocks in this appendix are the functional hierarchical blocks initially designed for use at the lowest levels of BLII. These blocks each perform a specific function relating to the processing of network traffic at the packet level. These blocks were removed from the model to improve the performance after the high level redesign. They are included for informational purposes. The following hierarchical blocks are contained in this appendix:

- Check Packet Format
- Packet Collection
- Find Queue Index
  - Jolt
  - Prioritize Next In
  - Process Exiting
  - Check For Match
  - Available Queue Process
  - Check For End
  - End of Search
- Sort Packets
- Bucket
- Convert to Packets
- Logical OR (Multiple)

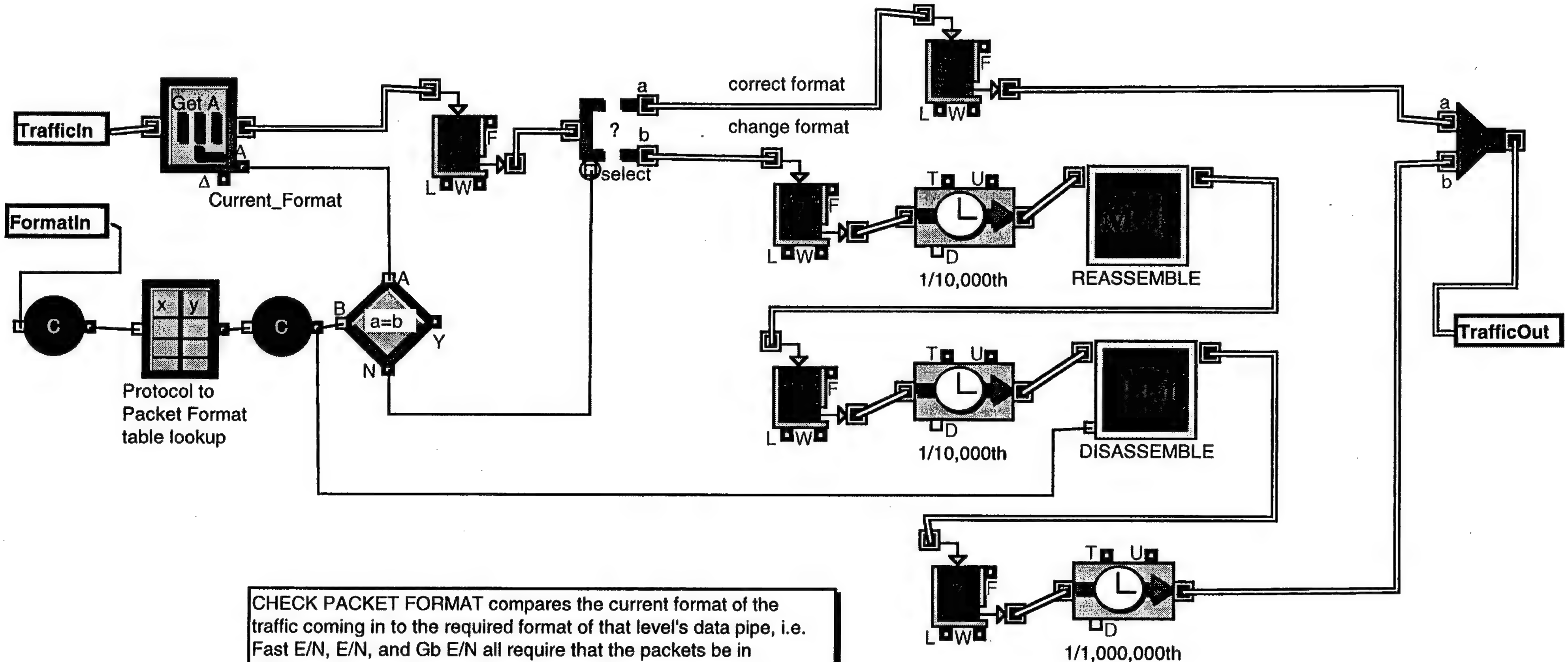
Structure of Check Packet Format

Icon of block Check Packet Format



Structure of Check Packet Format (COMPARE FORMAT PARTS.LIX)

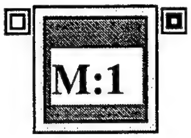
COMPARE THE CURRENT FORMAT OF THE INCOMING TRAFFIC TO DETERMINE IF IT IS IN THE FORMAT REQUIRED FOR THIS LEVEL TRANSPORT PIPE. IF NOT, CONVERT TO THE RIGHT FORMAT.



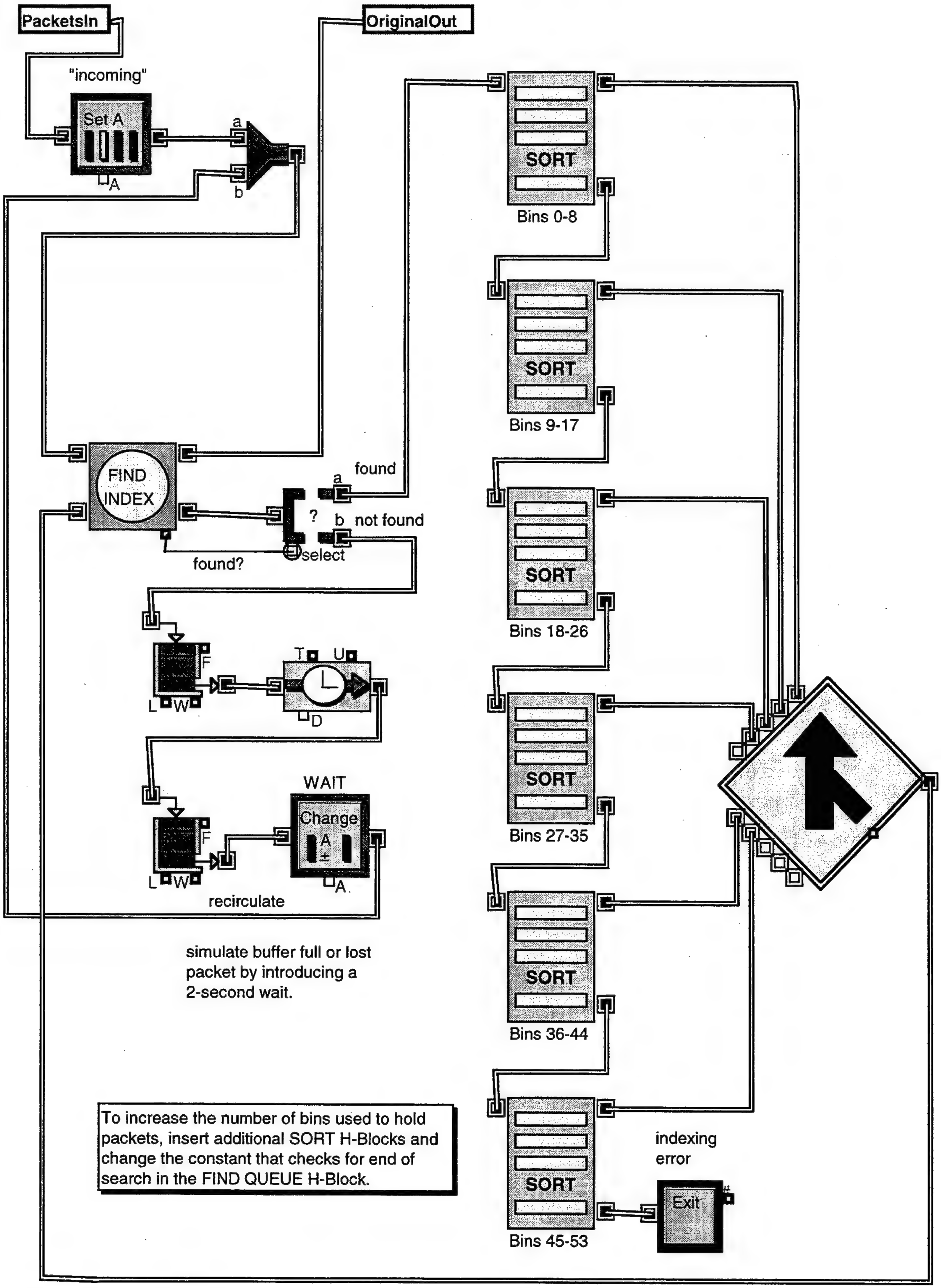
CHECK PACKET FORMAT compares the current format of the traffic coming in to the required format of that level's data pipe, i.e. Fast E/N, E/N, and Gb E/N all require that the packets be in Ethernet frames. Packets that are not in the correct format are collected until all have arrived, reassembled into the original format, then broken down into the required format for the current level.

Structure of Packet Collection (COMPARE FORMAT PARTS.LIX)

Icon of block Packet Collection



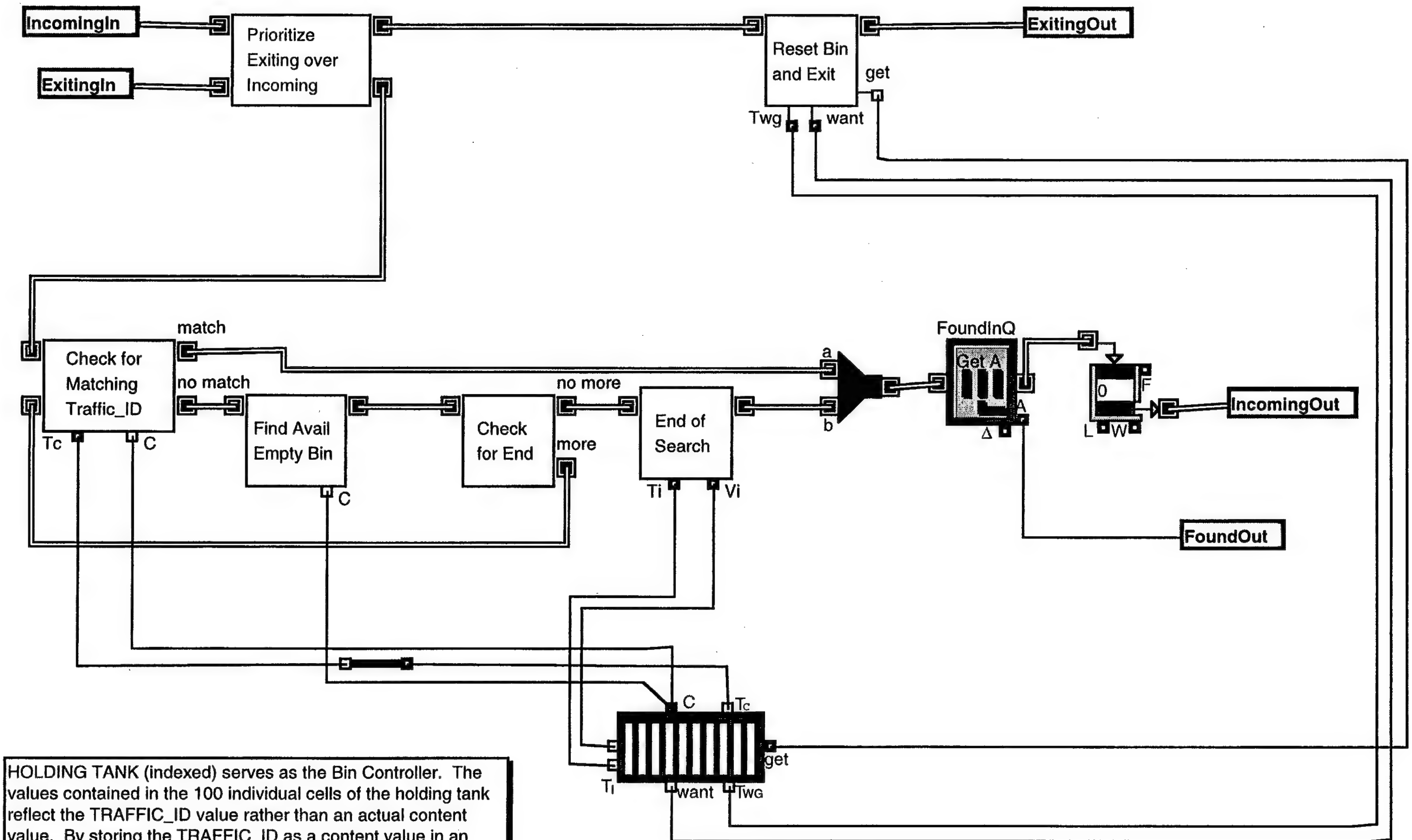
Structure of Packet Collection (COMPARE FORMAT PARTS.LIX)



Structure of Find Queue Index (ATMNETWORK.LIX)

Icon of block Find Queue Index

(9651)(6) Find Queue Index

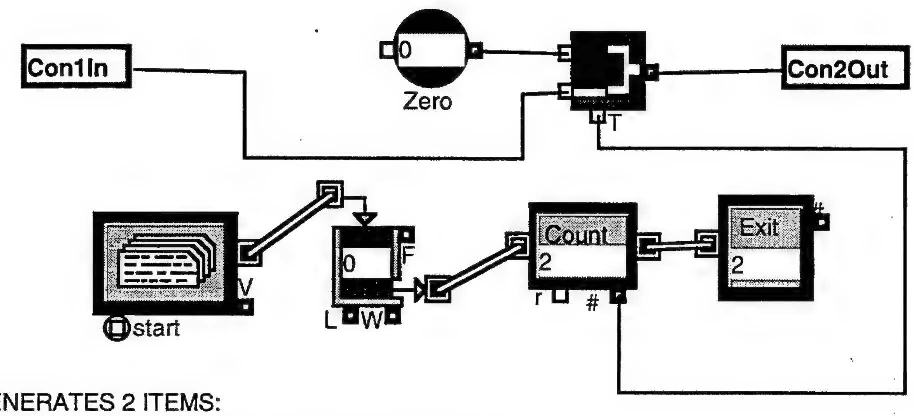


HOLDING TANK (indexed) serves as the Bin Controller. The values contained in the 100 individual cells of the holding tank reflect the TRAFFIC_ID value rather than an actual content value. By storing the TRAFFIC_ID as a content value in an independent cell, the holding tank can be searched for a cell holding the "right amount", thereby providing the Queue Index for the actual Bin that is collecting the packets for each unique TRAFFIC_ID.

Structure of Jolt (NETWORKUTILITIES.LIX)

Icon of block Jolt

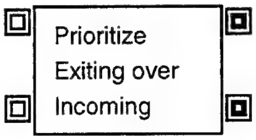
Structure of Jolt (NETWORKUTILITIES.LIX)



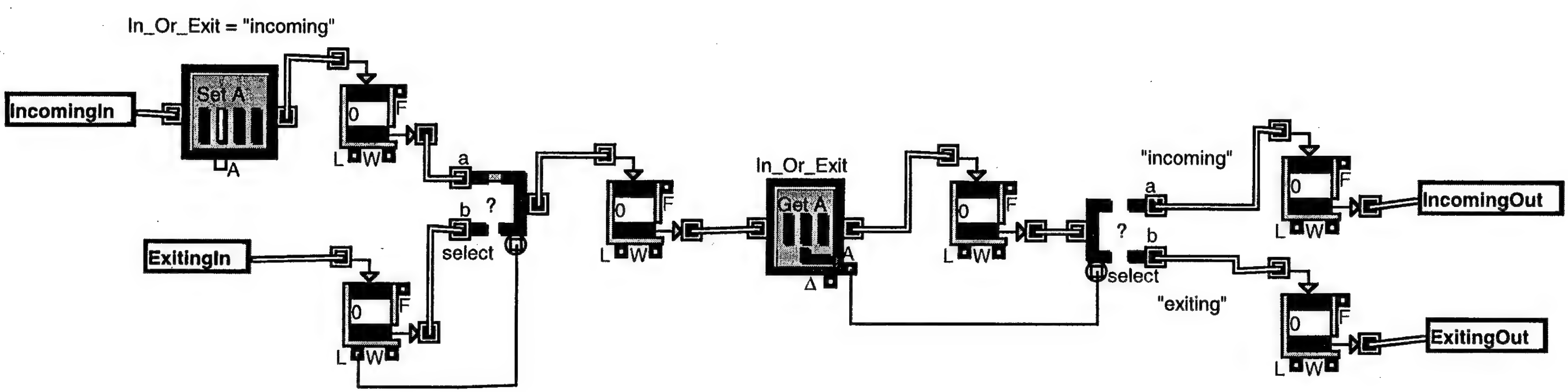
GENERATES 2 ITEMS:
Time=0 activates the Select Input to constant Zero.
Time=0.00000000001 generates last item to set the Select Input to normal values for remainder of the model run.

Structure of Prioritize Next In

Icon of block Prioritize Next In



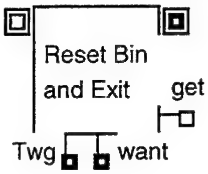
(9560)(0) Prioritize Next In



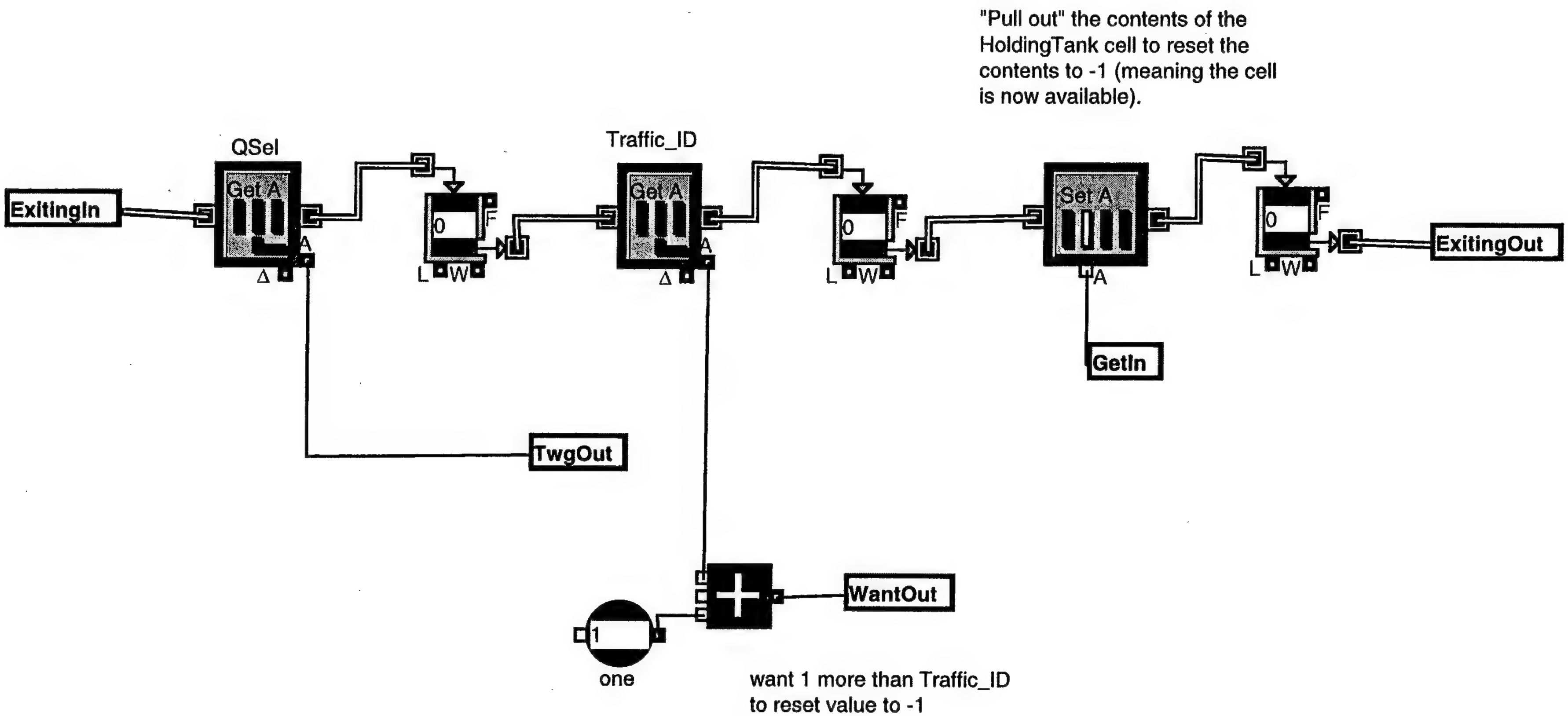
Select all "exiting"
traffic before
"incoming" traffic.

Structure of Process Exiting

Icon of block Process Exiting

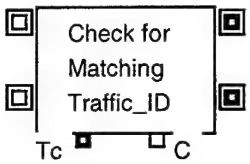


(9564)(4) Process Exiting

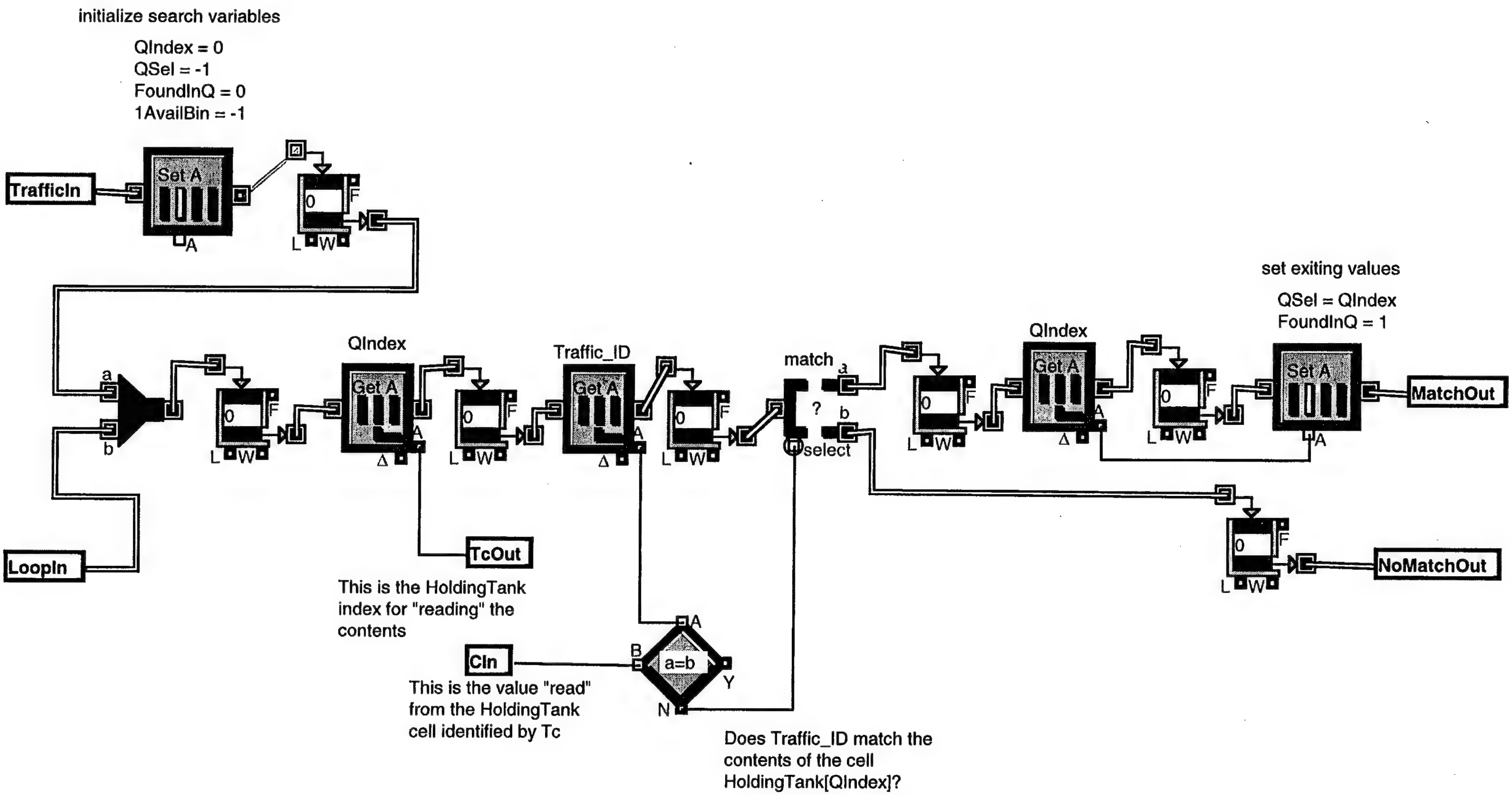


Structure of Check For Match

Icon of block Check For Match

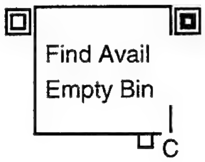


(9577)(17) Check For Match

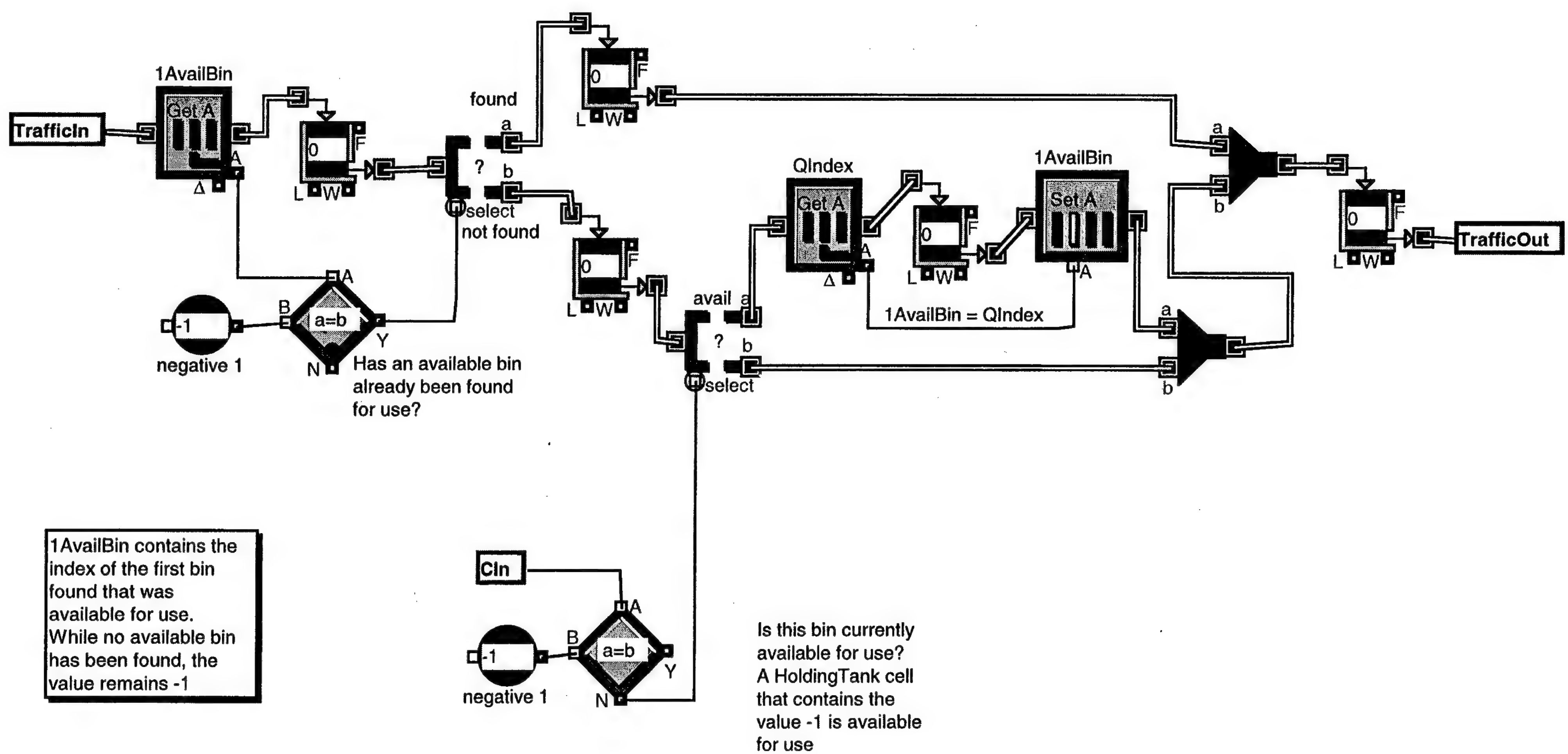


Structure of Avail Queue Process

Icon of block Avail Queue Process

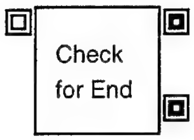


(9581)(21) Avail Queue Process



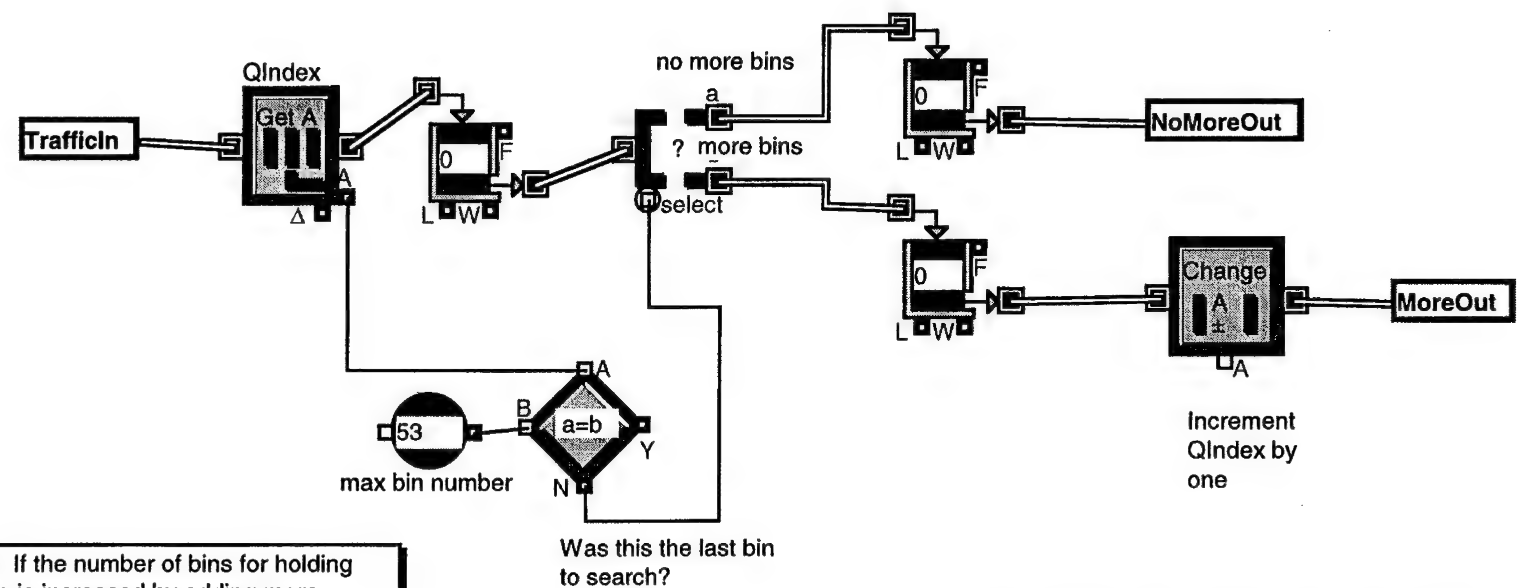
Structure of Check for End

Icon of block Check for End



(9562)(2) Check for End

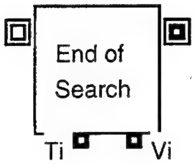
This block checks the current QIndex to see if it equals the largest bin index. If it does, then the item exits. If not, the QIndex is incremented by 1 and the item loops back to check the contents of the next HoldingTank cell.



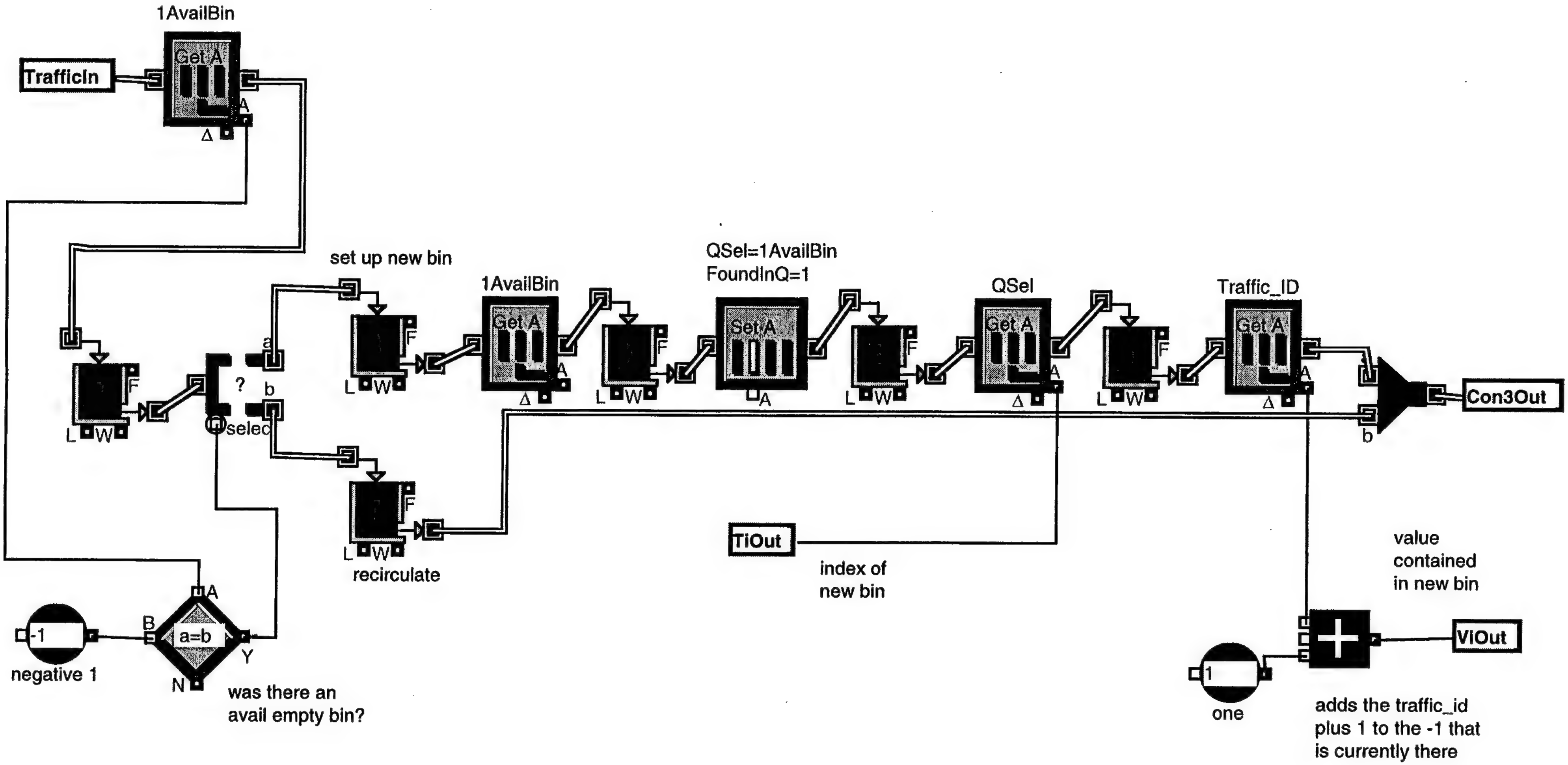
NOTE: If the number of bins for holding packets is increased by adding more SORT PACKET blocks, the constant identified by MAX BIN NUMBER must also be increased. The current configuration is for 54 bins, numbered 0 through 53. Max number allowed by the HoldingTank is 100.

ALSO NOTE: the number of bins being used to sort packets can be REDUCED to improve speed without having to actually reduce the number of SORT PACKET blocks. To do this, simply reduce the value in the MAX BIN NUMBER constant block.

Icon of block End of Search



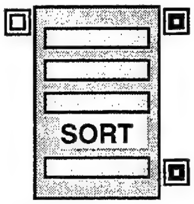
Structure of End of Search



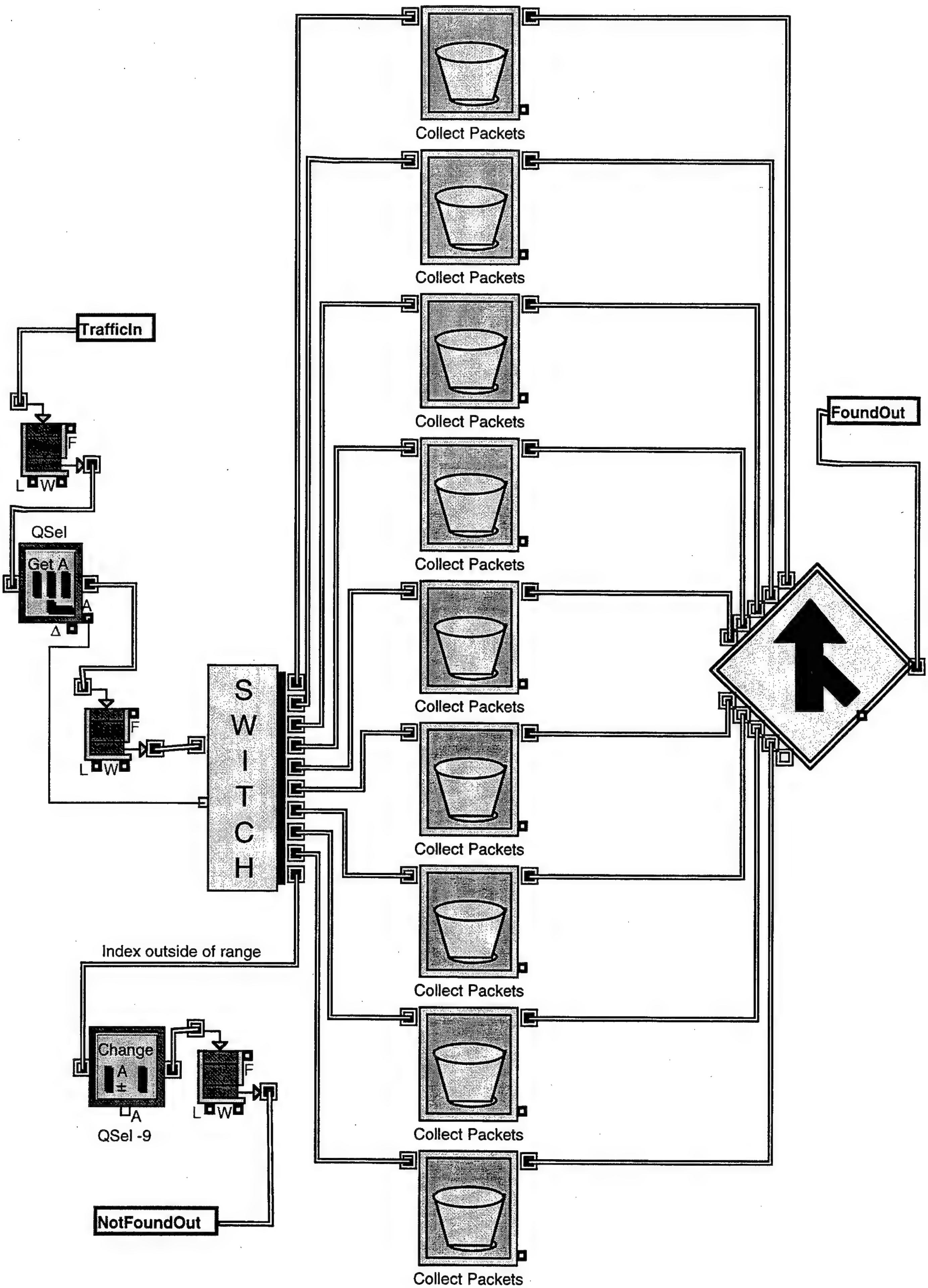
End of Search - 1

Structure of Sort Packets (COMPARE FORMAT PARTS.LIX)

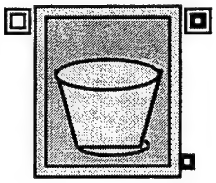
Icon of block Sort Packets



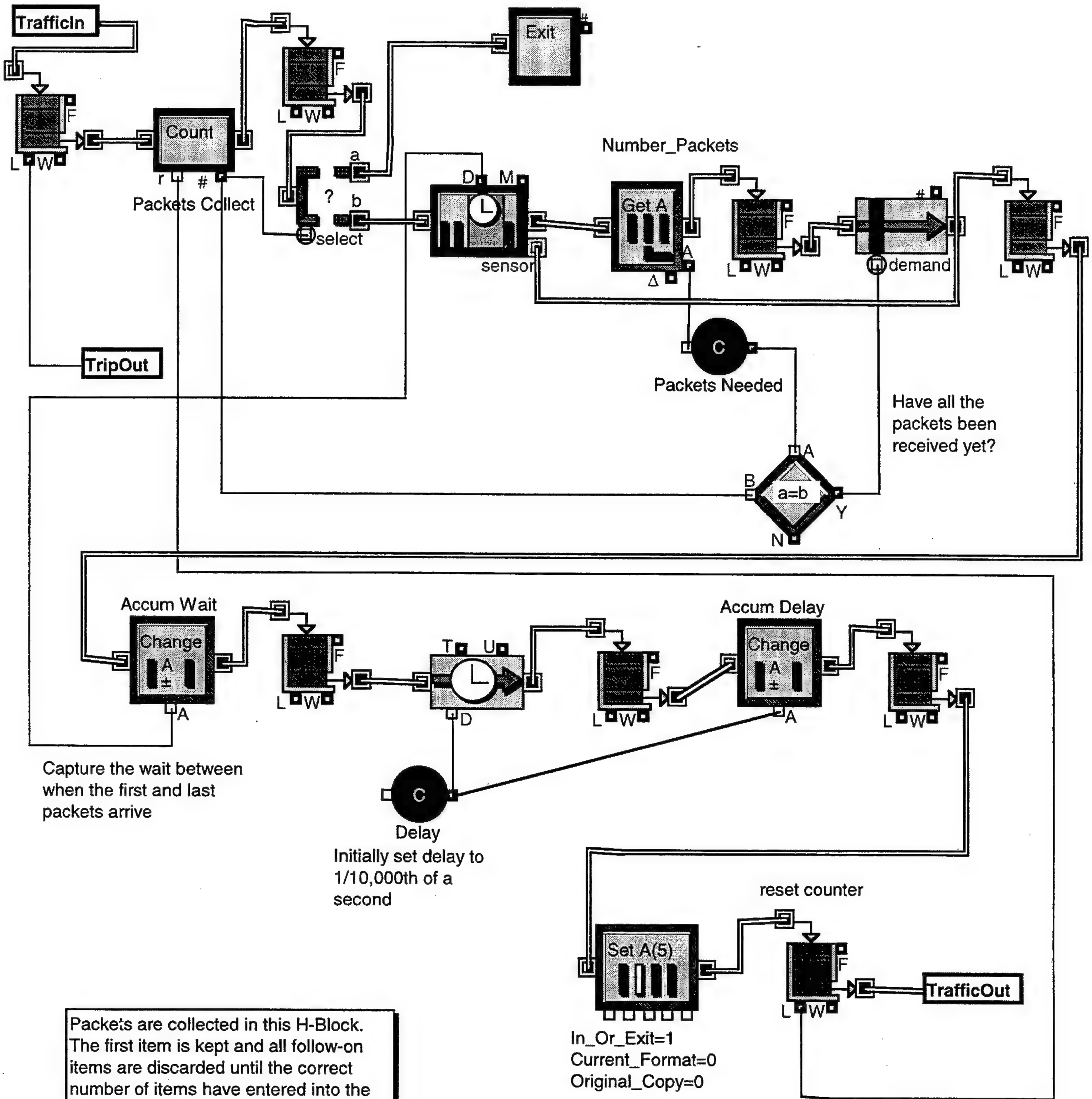
Structure of Sort Packets (COMPARE FORMAT PARTS.LIX)



Icon of block Bucket



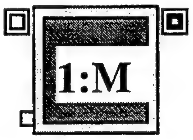
Structure of Bucket (COMPARE FORMAT PARTS.LIX)



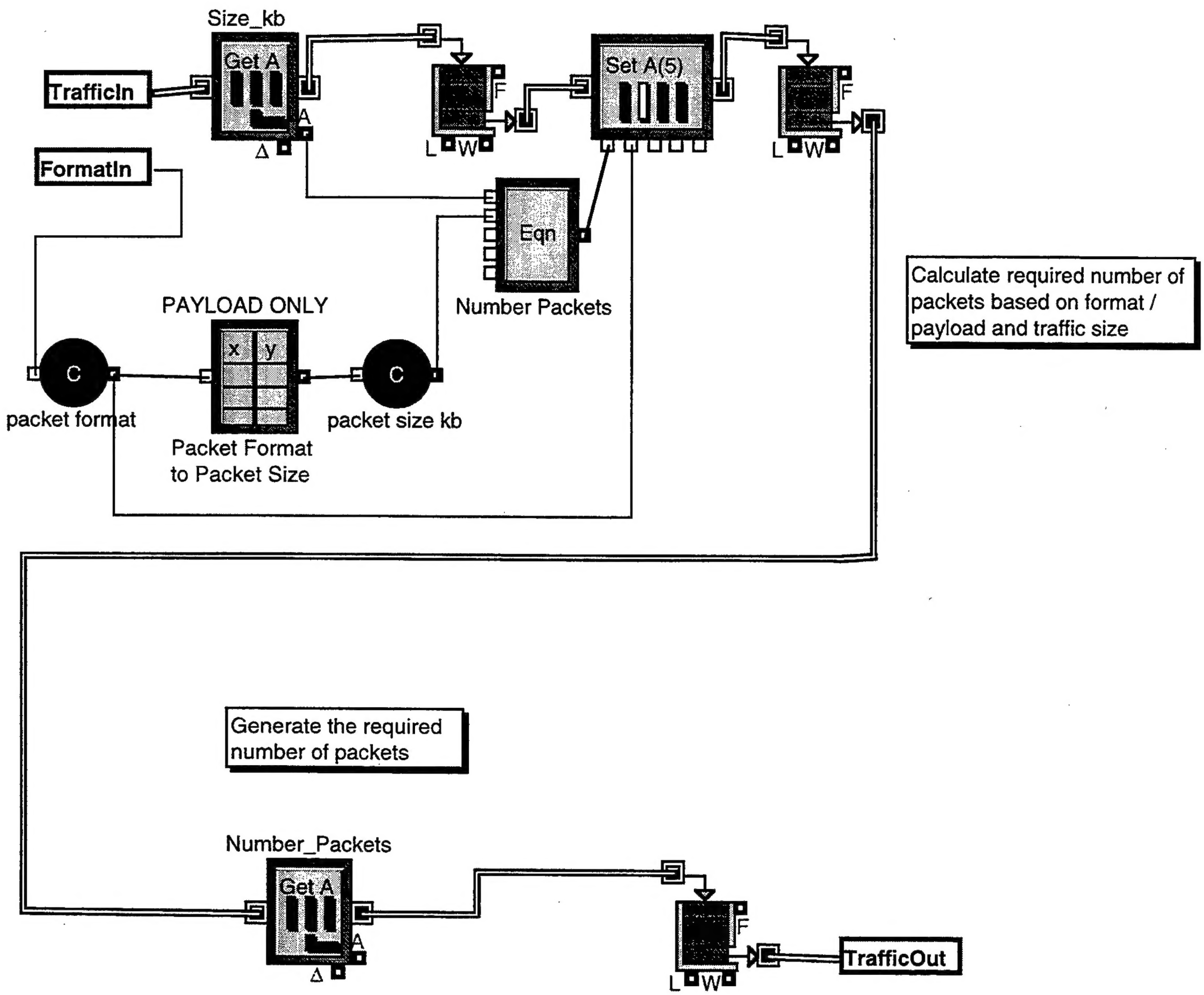
Packets are collected in this H-Block. The first item is kept and all follow-on items are discarded until the correct number of items have entered into the bucket. At that time, the packet is set to being re-assembled in its original traffic format.

Structure of Convert to Packets (COMPARE FORMAT PARTS.LIX)

Icon of block Convert to Packets



Structure of Convert to Packets (COMPARE FORMAT PARTS.LIX)



INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
   8725 John J. Kingman Road, Suite 0944
   Fort Belvoir, Virginia 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Road
   Monterey, California 93943-5101

3. Director, Training and Education 1
   MCCDC, Code C46
   1019 Elliot Road
   Quantico, Virginia 22134-5027

4. Director, Marine Corps Research Center 2
   MCCDC, Code C40RC
   2040 Broadway Street
   Quantico, Virginia 22134-5107

5. Director, Studies and Analysis Division 1
   MCCDC, Code C45
   300 Russell Road
   Quantico, Virginia 22134-5107

6. Marine Corps Representative 1
   Naval Postgraduate School
   Code 037, Bldg. 234, HA-220
   699 Dyer Road
   Monterey, California 93943

7. Marine Corps Tactical Systems Support Activity 1
   Technical Advisory Branch
   Attn: Major J.C. Cummiskey
   Box 555171
   Camp Pendleton, California 92055-5080

8. Professor William J. Haga, Code SM/HG 1
   Naval Postgraduate School
   Monterey, CA 93943

9. Professor John Osmundson, Code CC/OS 1
   Naval Postgraduate School
   Monterey, CA 93943

10. Major Gregory T. Gdanski 3
110 Shubrick Road
Monterey, California 93940